INTEGRATED COMPUTER SYSTEMS
5800 Hannum Avenue, P.O. Box 3614 / Culver City, CA 90231-3614

BULK RATE
U.S. POSTAGE
PAID
Integrated
Computer Systems

5735

TED NELSON

PROJECT XANADU
8480 FREDERICKSBURG 138
SAN ANTONIO TX 78229

May 1988

Dear Colleague:

An ongoing investment in enlarging your personal knowledge base is the key to enhancing your technical and managerial skills, advancing your career, and improving your ability to contribute - to make a difference.

Are you . . .

- being assigned new career challenges involving technology or management?
- in need of improved knowledge or skills in telecommunication, software, or electronic technology?
- in need of better methods for software development?
- involved in a project that uses state of the art hardware or software technology?

Then you will benefit by participating in one of our advanced technology courses. These courses are:

- **Practical** --you can immediately implement the information at your workplace,
- **Thorough** -- with workshop and hands-on formats, you'll learn the "How, as well as the What, and Why" of your subject,
- **Expertly taught** -- our instructors are experts in their field who are carefully chosen for their knowledge as well as teaching abilities,
- **Oriented to your needs** -- the class size is limited so the instructor can spend time on your particular application.

These courses work, and work well. The enclosed brochures give you detailed outlines for each of the 39 courses in our updated 1988 program. You'll find a selection of locations and dates from

- OVER -

which to choose. Enrollment details are on the back of the brochures. Join the ranks of over 150,000 computer and engineering professionals worldwide who have benefited from our courses. NOTE: If you'd like to speak with someone who has taken a course, we will be pleased to provide references.

Make a commitment today to invest in your personal knowledge base. Pick the right course for your personal growth and call us toll free today at (800) 421-8166.

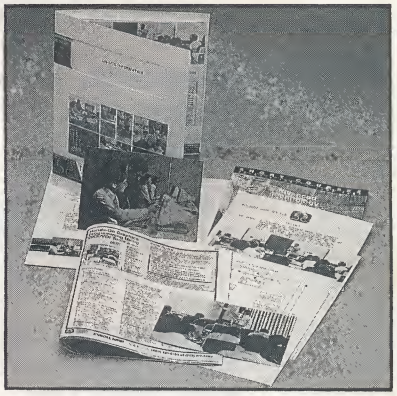
Sincerely,

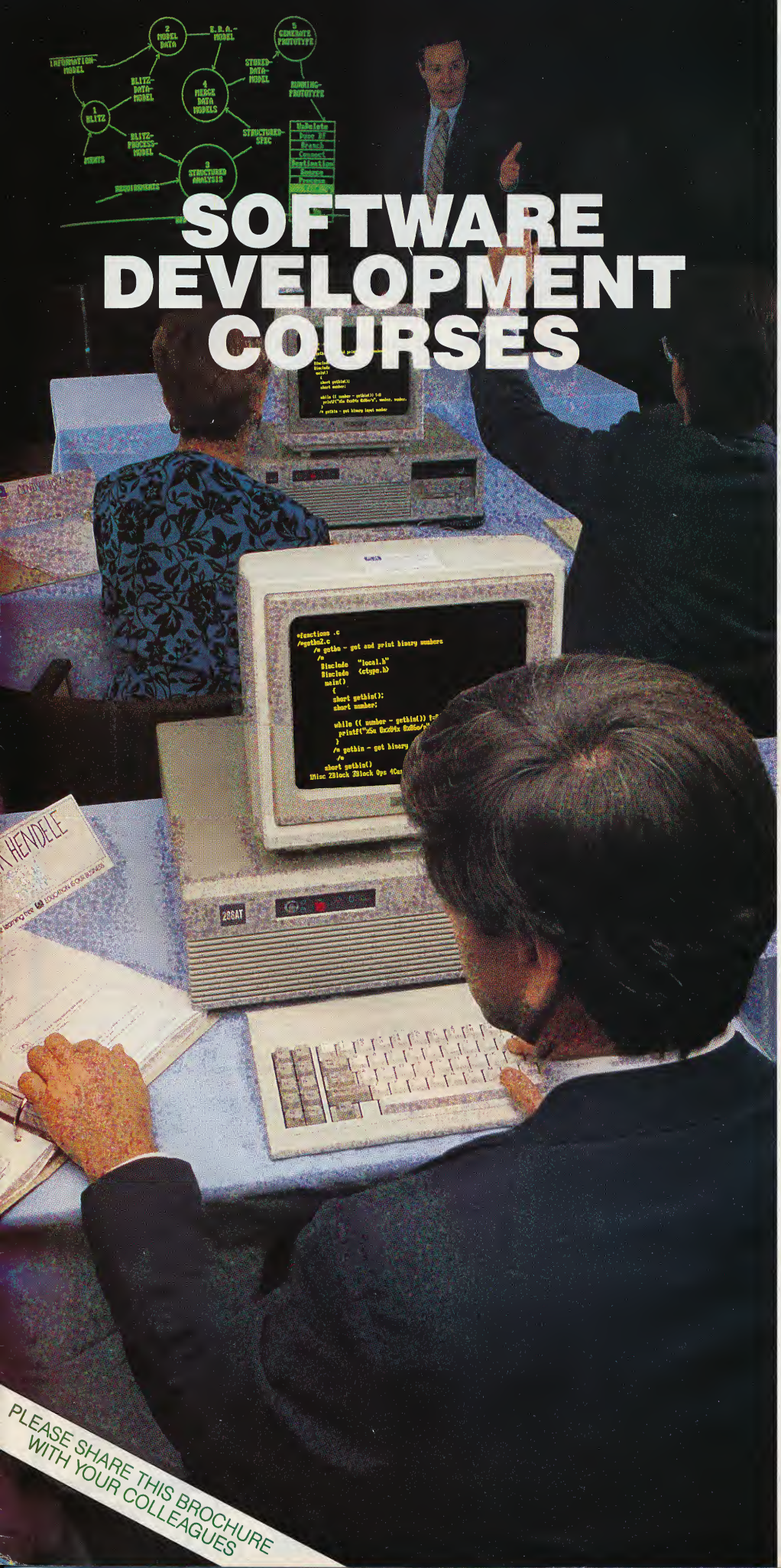
Eric R Garen

Eric. R. Garen
Executive Vice President

P.S. **100 SCHOLARSHIPS TO BE AWARDED:** To celebrate the opening of our Washington D.C. Education Center, we are offering you the chance to win one of 100 scholarships. This exceptional opportunity includes an expense-paid trip to Washington D.C. and complimentary attendance to the Integrated Computer Systems course of your choice in our new Washington Education Center. To enter your name in our drawing, return the enclosed postage paid reply card today or call us at (800) 421-8166.

P.P.S. Are there several people in your organization who have a need to acquire the same knowledge and skills? We can provide an "on-site" course at your facility, including instructor, course materials, and all required hardware and software. Just call for a FREE "On-Site Information Package" which provides you with information on how to arrange a cost-effective, on-site course adapted to your specific needs.





PLEASE SHARE THIS BROCHURE WITH YOUR COLLEAGUES

SHORT COURSES

May 1988-December 1988

Languages & Operating Systems

- ► C Programming: Hands-On Workshop HANDS-ON 2
- **NEW!** Advanced C Programming: Hands-On Workshop HANDS-ON . . . 3
- ► Ada® Programming & Software Engineering: Hands-On Introduction HANDS-ON 4
- ► UNIX Hands-On Workshop HANDS-ON 5

Software Engineering

- **NEW!** Software Requirements & Specifications 6
- **NEW!** Software Quality Assurance & Testing 7
- **NEW!** Software Maintenance 8
- ► Software/User Interfaces: Design Principles & Techniques DEMO LABS 9
- ► Structured Design & Programming 10
- ► CASE: Computer-Aided Software Engineering, Hands-On HANDS-ON 11
- ► Computer Graphics: Software Techniques & Hardware Components DEMO LABS 12
- ► Relational Databases: Design, Tools & Techniques DEMO LABS 13
- ► Real-Time System Design: A Hands-On Workshop HANDS-ON 14

Enrollment & On-Site Courses 16

from the learning tree®

INTEGRATED COMPUTER SYSTEMS

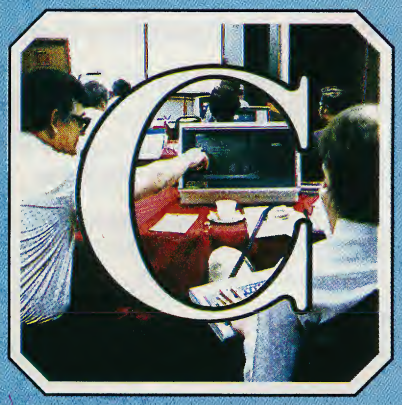
EDUCATION IS OUR BUSINESS®

8805UB

C Programming: Hands-On Workshop

HANDS-ON

Course 338



4 Days

WASHINGTON, D.C.
June 21-24
SAN DIEGO
June 28-July 1
BOSTON
July 12-15
WASHINGTON, D.C.
July 26-29
LOS ANGELES
August 16-19
WASHINGTON, D.C.
August 23-26
PALO ALTO
September 13-16
WASHINGTON, D.C.
September 20-23
SAN DIEGO
October 4-7

Courses at Your Facility
Please Inquire

8805 UB 338

Course Benefits

The best way to learn a language is to have hands-on experience under the guidance of an expert instructor. Throughout the course, participants use UNIX/C workstations for extensive hands-on exercises in writing and executing programs in the C Language.

The C Language has become a key tool for fully exploiting the speed and power of the new generation of micro- and minicomputer systems. C offers the structure and portability of a High-Level Language while retaining the detailed machine control, efficiency, and speed of assembly language. The popularity of C for engineering and systems programming applications is due to its simple structure, ease of use, applicability to real-time systems, and extensive library of functions.

YOU WILL LEARN HOW TO:

- **Write, Compile, and Execute C Programs**
- **Make Full Use of C's Powerful Structured Programming Features**
- **Implement C Programs for Real-Time Applications**
- **Link Assembly Language Modules to C Programs**
- **Develop Programs Rapidly Using C Library Functions**
- **Use Software Tools for C Development**

WHO SHOULD ATTEND:

This course is valuable for engineers, scientists, applications and systems programmers, systems analysts and managers who plan to use C for either system or application programming. Participants should have some background in programming, using either high-order or assembly language.

Course Content

1. INTRODUCTION & OVERVIEW

Fundamental Concepts & Terminology
- Evolution of C from ALGOL & BCPL
- C & the UNIX Operating System
- Comparing C with Other Languages
- Micro, Mini & Mainframe Availability of C

Key Features of C
- Operating System Independence & Portability
- Highly Efficient Object Code
- Ease of Access to Hardware Features
- Support of Structured Programming
- Separate Compilation of Modules
- Extensive Libraries of Functions

Applications of the C Language
- Engineering & Real-Time Applications
- Graphics & Database Applications
- Systems Programming: Languages, Operating Systems & I/O Drivers

2. USING UNIX TO PREPARE C PROGRAMS

Introduction to the UNIX System
- Short List of Useful Commands
- Preparing Files with the **vi** Text Editor
- Moving, Copying, & Removing Files

Writing a Simple C Program
- Learning the Form of a C Program
- Declaring Variables
- Designing Program Flow & Control
- Using Procedures & Functions
- Implementing Terminal Input & Output

3. FUNDAMENTALS OF C PROGRAMS

Fundamental Types & Storage Classes
- Character Types
- Integer: Short, Long, Unsigned
- Single- & Double-Precision Floating Point
- Storage Classes: Automatic, Register, Static, & External

Operators & Expressions
- Using Numeric & Relational Operators
- Mixed Operands & Type Conversion
- Conditional Operators ■ Bit Operations
- Operator Precedence & Associativity

Conditional Program Execution
- Applying **if** & **switch** Statements
- Nesting **if** & **else**
- Restrictions on **switch** Values
- Use of **break** & **default** with **switch**

Program Loops & Iteration
- Uses of **while** & **do** Statements
- When to Use **for** Statements
- Using **break** & **continue**

4. FUNCTIONS, POINTERS, & ARRAYS

Modular Programming
- Passing Arguments by Value & Reference
- Scope Rules & Global Variables
- Separate Compilation & Linkage
- Building Your Own Function Library
- Portability Considerations

Arrays
- Array Notation & Representation
- Manipulating Array Elements
- Using Multidimensional Arrays
- Arrays of Unknown or Varying Size

Structures
- Purpose & Usage of Structures
- Declaring Structures
- Assigning Values to Structure Variables
- Nested Structures & Arrays of Structures

Pointers to Objects
- Graph Structures & Pointers
- Pointer & Address Arithmetic
- Pointer Operations & Declarations
- Using Pointers as Function Arguments

Packaging of Data Structures
- Keeping Data Structures in Separate Files
- Public **.h** vs. Private **.c** Files
- Hiding Private Variables & Routines

5. SYSTEM PROGRAMMING TECHNIQUES

File System I/O Techniques
- Formatted I/O
- Reading & Writing of Structures
- Access to Named Files
- Sequential & Non-Sequential File Access

Use & Control of Parallel Processes
- Forking & Invoking Processes
- Using Pipes & Signals
- Real-Time Handling of Interrupts

Controlling I/O Ports & Devices
- System I/O Drivers
- Bit Access & Masking
- Communication Interfaces

Linking to Assembly & Foreign Code

6. SOFTWARE TOOLS FOR C PROGRAMMING

The Standard C Preprocessor
- Defining & Calling Macros
- Utilizing Conditional Compilation
- Passing Values to the Compiler

Debugging C Programs
- Automatic Checking of Usage & Compatibility with **lint**
- UNIX Tools for Symbolic Debugging
- Building Debugging Support into Programs

Additional C Software Support
- Creating Parsers with **yacc**
- Profiling C Programs
- The C Beautifier & Cross-Referencer
- Standard String & I/O Packages
- Other Standard Library Packages

7. CURRENT C IMPLEMENTATIONS

UNIX & Independent C Compilers
Recent Developments in the C Language

HANDS-ON

This course is unique in providing participants guided Hands-On experience with significant C programming exercises throughout the course.



INTEGRATED COMPUTER SYSTEMS

ENROLL NOW!

Call Toll Free **(800) 421-8166** or

East Coast Office **(703) 847-6448**

West Coast Office **(213) 417-8888**

8805 UB 339

Advanced C Programming: Hands-On Workshop

HANDS-ON

Course 339

NEW!



4 Days

PALO ALTO
May 24-27
WASHINGTON, D.C.
June 7-10
SAN DIEGO
July 12-15
BOSTON
July 19-22
WASHINGTON, D.C.
August 2-5
LOS ANGELES
September 20-23
WASHINGTON, D.C.
September 27-30
BOSTON
October 25-28
SAN FRANCISCO
November 15-18

Courses at Your Facility
Please Inquire

Course Benefits

C has become an extremely popular programming language for advanced applications, system's programming, sophisticated user interfaces and state-of-the-art software techniques. The wide availability of powerful C compilers, function and program libraries, and the portability of C code, has led to its adoption as the language of choice for professionals using minis, workstations and PCs.

In this course extensive **hands-on exercises using Standard C, conducted under the guidance of an expert instructor**, are used to guide the participant in the use of advanced algorithms, the construction of complex data structures and the design and maintenance of very large C programs.

YOU WILL LEARN HOW TO:
- **Design and Implement Advanced Algorithms in C**
- **Select and Customize Complex Data Structures**
- **Exploit Advanced Techniques and C Development Tools**
- **Structure, Modify and Debug Large, Multi-Module C Programs**
- **Generate Reliable, Portable and Maintainable Software**
- **Create Function Libraries for Specific Application Needs**

WHO SHOULD ATTEND:

This course is valuable for engineers, programmers, analysts, or technical managers who are responsible for the development and/or maintenance of advanced application or system programs in C. Hands-on experience with the C language at the level of Course 338, "Hands-On Programming in C," or the equivalent, is assumed.

Course Content

1. AN OVERVIEW OF C ENVIRONMENTS

The Edit/Compile/Debug Cycle
- Editors Which Support C Programming
- Symbolic Debuggers and Interpreters
- Modularity ■ Separate Compilation

Software Tools for C Programming
- Using **make** to Manage Dependencies
- Controlling Versions and Revisions
- Static Debugging with **lint**

Extensive Libraries of C Functions
- The Extended Arithmetic Library
- Libraries for Menus, Windows and Graphics
- Building Your Own Application Libraries

2. DATA STRUCTURES AND ALGORITHMS

Advanced String and Text Processing
- Automatic Allocation for Dynamic Strings
- Simulating Virtual Memory and Paging
- Implementing Text Buffers and Markers

Recursive Algorithms
- Choosing Between Recursion and Iteration
- Implementing Quicksort in C
- Binary and Balanced Trees
- Backtracking

Dynamic Data Structures
- Implementations of Sparse Arrays
- Writing Macros for Multiword Bitsets
- Singly and Doubly Linked List Processing
- Implementing LIFO Stacks and FIFO Queues

3. USING AND BUILDING LIBRARIES

Using Existing Libraries
- The ANSI Standard C Library
- Public Domain Libraries
- Commercial C Libraries

Building Application Support Libraries
- Packaging Macros and Data Types
- Declaring Public and Private Functions
- Referencing External Variables

4. PROGRAM DESIGN AND DEBUGGING

Designing Modular Programs
- Accepted Software Engineering Practices
- Designing Clean Interfaces ■ Prototyping
- Inserting Test Code for Bug Detection

Programming for High Reliability
- Assertions and Loop Invariants
- Checking Recursive Functions
- Controlling Pointers and Indices

Programming for Efficiency
- Indexing vs. Address Arithmetic
- Using Multiple Indirection
- Bit-Structures and Unions

5. OBJECT ORIENTED PROGRAMMING

Data Driven Programming
- Attaching Functions to Data
- Finite State Machine Algorithms
- Functions Which Return Functions

Full Object Oriented Programming
- Declaring Object Instances and Classes
- Sending Messages to Objects
- Providing for Multiple Inheritance

Object Oriented Preprocessors
- Macros for Message Passing
- The *Objective C* Language
- The *C++* Language

6. USING C IN LARGE PROJECTS

Project: An Extensible Text Editor

Controlling the Display
- Controlling Terminal Features
- Managing Multiple Windows
- Accessing the Screen Memory

Command Processing
- De-Multiplexing and Parsing Terminal Input
- Recovering from User Errors
- Adding Macros and Abbreviations

Managing Files
- Providing Transaction Security
- Generating Audit Trails
- Implementing Support for Version Control

7. CONCURRENT PROGRAMMING

Semi-Concurrent Programming
- Processing Interrupts in C
- Using Long Jumps and Returns
- Implementing Multiple Co-routines

General Techniques
- Creating Parallel Processes
- Binary and Counting Semaphores
- Interprocess Communication

Parallel Processing Under UNIX
- Using **fork** and **exec** ■ System STREAMS

Parallel Processing Under MS-DOS
- Commercial Real-Time OS Kernels
- Multi-tasking Under MS-DOS 5.0

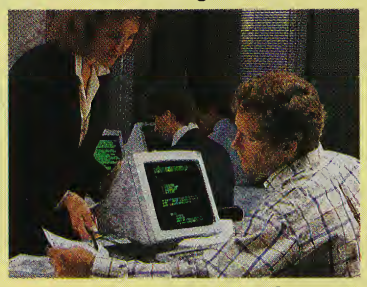
8. SPECIAL PURPOSE TECHNIQUES

Techniques for Device Drivers
- Control of Memory Mapped Devices
- Interfacing with Assembly Language
- Creating Reloadable Programs

Generating Portable Code
- Using Macros vs. Conditional Compilation
- Effects of Word Size, Bit and Byte Order
- Virtual Operating System Calls

HANDS-ON

An essential element of the course is the use of Hands-On Exercises in Advanced C programming, including:
- **Using Advanced Software Tools**
- **Developing Powerful Library Functions**
- **Maintaining A Large C Program**
- **Implementing Dynamic Data Structures**
- **Object Oriented Programming**
- **Parallel Processing in C**



Participants develop Advanced C programs under the guidance of an expert instructor.

INTEGRATED COMPUTER SYSTEMS

ENROLL NOW!

Call Toll Free **(800) 421-8166** or

East Coast Office **(703) 847-6448**

West Coast Office **(213) 417-8888**

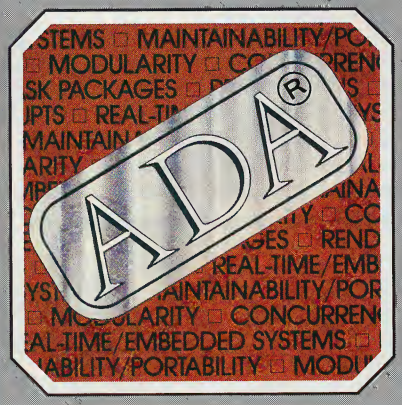
Ada® is a registered trademark of the Department of Defense (AJPO).

Ada® Programming & Software Engineering: Hands-On Introduction

8805 UB 335

HANDS-ON

Course 335



4 Days

SAN DIEGO
June 21-24
WASHINGTON, D.C.
June 28-July 1
WASHINGTON, D.C.
August 9-12
LOS ANGELES
August 30-September 2
WASHINGTON, D.C.
September 13-16
SAN DIEGO
September 27-30
BOSTON
October 4-7
SAN FRANCISCO
November 29-December 2

Courses at Your Facility
Please Inquire

Course Benefits

Validated, production quality Ada compilers are now available for a wide variety of computers. Ada has become the "programming language of choice" and a principal vehicle for implementing modern software practices in applications ranging from large, complex systems to real-time, embedded environments.

The best way to learn Ada is to gain hands-on experience under the guidance of an expert instructor. Participants use In-Class workstations throughout the course for extensive hands-on exercises in designing and testing Ada programs using a validated Ada compiler.

YOU WILL LEARN HOW TO:

- **Make Full Use of Ada's Modern Software Engineering Features**
- **Use Ada's Modular Packages in System Design and Implementation**
- **Write, Compile, Link and Execute Ada Programs**
- **Employ Information Hiding and Data Abstraction**
- **Develop Concurrent Programs Using Ada's Task Facility**
- **Interface to Code Written in Other Languages**

WHO SHOULD ATTEND:

This course is valuable for programmers, systems analysts, engineers, and technical managers who will be using Ada for the design and implementation of application or systems programs. Participants should have a background in programming using a high-level language such as Pascal, C or FORTRAN.

Course Content

1. INTRODUCTION & OVERVIEW

Ada's Goals & Software Engineering
- A Language and a Design Tool
- Facilities for Structured Programming
- Object-Oriented Design
- Maintainability & Portability
- Reliability, Understandability
- Support for Concurrent Programming

Ada Language Features
- Strong Typing & Data Abstraction
- High-Level Access to Hardware Resources
- Information Hiding & Encapsulation
- Libraries for Modular, Reusable Software

2. HANDS-ON INTRODUCTION TO ADA

Using the In-Class Computer System
- Simple Operating System Commands
- Creating & Editing Ada Programs
- Moving, Copying, & Removing Files

Writing an Ada Program
- Packages, Declarations & Statements
- Compiling & Running an Ada Program
- Testing & Debugging the Program

3. ADA FUNDAMENTALS

Structure of Ada Program Units
- Top Level View of Ada Program Organization
- Lexical Building Blocks
- Using Packages for Modularity

Subprograms: Procedures & Functions
- Parameters & Parameter Modes
- Internal Structure

Objects and Types
- Variables & Constants ■ Attributes & Types
- Scalar & Composite Types

Expressions & Statements
- Names & Aggregates
- Operations & Expression Evaluation
- Statements: Assignment, Conditional, Loop, Exit

4. APPLYING ADVANCED DATA TYPES

Data Type Concepts
- Data Values & Operations
- Declaring Subtypes & Derived Types
- Type Checking & Conversion
- Standard Types: Integer, Float, Fixed Point
- Enumeration, Record, Array, & Access Types

Using Subtypes & Private Types
- Constraints, Record Variants & Discriminants
- Keeping Information Hidden
- Defining Private Types

5. TASKS & CONCURRENT PROGRAMMING

Concurrent Programming Concepts
- Structuring Concurrent Programs
- Logical vs. Physical Concurrency
- Design of Interprocess Communication
- Real-Time & Event-Driven Programs

Using Tasks & Task Types
- The Rendezvous Concept
- Entries: Callers & Accepters
- Avoiding Task Deadlock and Idle Tasks
- Using Delay Statements and Timeouts

Exception Handling
- When to Use Exceptions
- Declaring Exceptions
- Handling & Propagating Exceptions

6. BUILDING SYSTEMS WITH ADA

Developing Modular Programs
- Top-Down Design: Program Stubs
- Bottom-Up Program Construction
- Reusing Generic Modules
- Consistency Checking & Version Control

Organizing Program Components
- Independent vs. Separate Compilation
- Using Compiler Directives: Pragmas
- Compilation Units & Program Libraries
- Scope, Visibility, & Renaming
- Dynamic Memory Allocation

7. EXTERNAL INTERFACING

Input/Output & Data Representation
- Formatted I/O, Sequential & Direct Files
- Data Packing, Length, & Alignment
- Address Specification ■ Unsafe Conversion

Using "Foreign" Code & Hardware Devices
- Machine Code Insertions
- Linking to Subroutines in Other Languages
- Handling Interrupts
- Real-Time, Embedded Systems

8. SOFTWARE ENGINEERING TOOLS

Ada Compiler Implementations
- Validation of Ada Compilers
- Current Ada Implementations
- Tradeoffs in Choosing an Ada Compiler

Program Support Environments (APSE)
- Kernel, Minimal, and Extended APSEs
- The Host Machine & Operating System
- Managing Libraries & Configurations
- Tools for Program Development
- Host/Target Cross Compilers
- CAIS (Common Ada Interface Standard)
- DoD Standard 2167

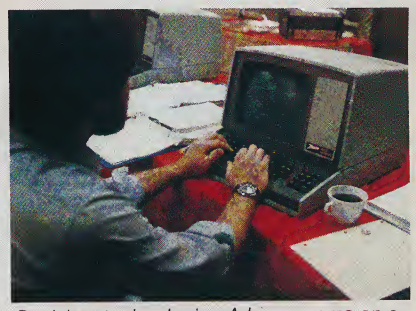
Software Development Considerations
- Impact on Software Project Management
- Using Ada as a Program Design Language
- Software Development Standards
- Ada Style Guidelines ■ Ada User Groups

HANDS-ON

Throughout the course, programming concepts and skills are reinforced by extensive use of hands-on experience.

Hands-on exercises include:

- **Writing Interactive Ada Programs**
- **Developing Generic Modules & Reusable Packages**
- **Implementing a Database in Ada**
- **Using Tasks and Exceptions**



Participants developing Ada programs on a multi-user system using a validated, interactive Ada compiler.

INTEGRATED COMPUTER SYSTEMS

ENROLL NOW!

Call Toll Free **(800) 421-8166** or

East Coast Office **(703) 847-6448**

West Coast Office **(213) 417-8888**

*UNIX is a registered trademark of AT&T.
Ada® is a registered trademark of the Department of Defense (AJPO).

8805 UB 336

UNIX* Hands-On Workshop

HANDS-ON

Course 336



4 Days

WASHINGTON, D.C.
June 14-17
BOSTON
June 28-July 1
PALO ALTO
July 19-22
LOS ANGELES
August 2-5
WASHINGTON, D.C.
August 16-19
WASHINGTON, D.C.
September 20-23
SAN DIEGO
September 27-30
BOSTON
October 25-28
LOS ANGELES
November 1-4

Courses at Your Facility
Please Inquire

Course Benefits

UNIX is a powerful multi-user, multi-tasking operating system with a rapidly growing number of users on a wide range of computers and workstations.

The best way to learn the UNIX operating system is to actually **USE IT.** This course provides **extensive Hands-On** training under the guidance of an expert instructor. Participants will use In-Class UNIX workstations for hands-on exercises throughout the course.

The course provides a comprehensive, hands-on introduction to the full range of UNIX facilities, services and utilities. It includes in-depth coverage of both the System V and BSD versions, and emphasizes the important UNIX tools for system software and applications development.

YOU WILL LEARN HOW TO:

- **Understand and Utilize the Key Facilities, Services and Utilities of the UNIX Operating System**
- **Employ UNIX Shells, Shell Scripts, Pipes and Filters**
- **Use the Programmer's Workbench for Software Development**
- **Perform Key Systems Integration and Maintenance Functions**
- **Extend & Customize the UNIX System for Specific Applications**
- **Make Use of UNIX Networks and Network Interfaces**

WHO SHOULD ATTEND:

This course is valuable for engineers, systems and applications programmers, systems analysts, software support specialists and managers who want to acquire a solid foundation in the UNIX operating system.

Course Content

1. INTRODUCTION & OVERVIEW

Fundamental Concepts & Terminology
- Basic Components: Kernel, Shells, System Utilities, the C Language
- Managing System Resources Efficiently
- UNIX Systems on Workstations & Micros
- Latest AT&T System V, BSD & Xenix Developments

Advantages of the UNIX Operating System
- Multi-User, Multi-Tasking Capabilities
- Machine & Terminal Independence
- Extensive Software Tools & Utilities
- Using UNIX with DOS, VMS and other Operating Systems

UNIX Applications
- Advanced Program Development
- Graphics ■ Scientific Computation
- Office Automation ■ Databases
- CAD/CAM Systems ■ Real-Time Control

2. USING THE UNIX FILE SYSTEM

UNIX File & Directory Features
- Ownership, Permissions, Access
- Hierarchical Directory Structure
- Absolute & Relative Path Names

UNIX Filesystem Conventions
- Filename Extensions & Regular Expressions
- Moving & Removing Files (Links)

3. THE UNIX SHELLS: UNDERSTANDING AND EXPLOITING THEIR POWER

Using the UNIX Shell
- Understanding the Command Interpreter
- Redirecting I/O ■ Using Pipes & Filters
- Forking a UNIX Process
- Shell Control Structures & Variables
- Using Shellscripts as Programs

Other UNIX Shells

4. UNIX EDITORS & DOCUMENTATION AIDS

UNIX Screen & Line-Oriented Editing
- Basic Screen-Oriented Editing with **vi**
- Advanced Editing Techniques with **ex**
- UNIX Text & Document Formatters

5. THE UNIX PROGRAMMING ENVIRONMENT

The Programmer's Workbench
- Automatic File Maintenance: **make**
- Version Control Systems: **sccs** & **rcs**
- A General Purpose Macroprocessor: **m4**
- The **yacc** Program Generator

Programming in C
- Basic Features & Advantages of C
- UNIX Tools for the C Language
- System Calls and Standard Libraries

FORTRAN & Pascal Tools
- **ratfor**: Rational FORTRAN Preprocessor
- Pascal Interpretation vs. Compilation Choice
- Cross Referencing & Execution Profiling

Other Languages Available
- Ada® ■ BASIC ■ APL ■ Prolog
- COBOL ■ LISP ■ Others

6. UNIX SYSTEM ADMINISTRATION

System Initialization
- Bringing the System Up & Shutting It Down
- Creating Accounts, Users & Groups

Performing System Maintenance
- Adding & Deleting Devices: **mknod**
- Modifying Terminal Characteristics: **ttys**
- Performing Back-ups & Restores: **dump, tar**

7. LOCAL & NETWORK COMMUNICATION

Local Area Network Resource Sharing
- File and Print Servers
- IPC and Network File System
- LAN Network Configurations

UNIX Networks & Network Interfaces
- Ethernet & PABX-Based UNIX Networks
- The **uucp** UNIX to UNIX Copy Network
- Using Public Network Gateways

8. IMPLEMENTATIONS & PORTABILITY

Hardware Requirements
- Word & Address Size
- Memory Management
- Memory & Disk Capacity/Latency

Porting UNIX Systems to New Processors
- Changes to the Kernel & C Compiler
- Customizing UNIX for Special Applications

UNIX Dialects & Support
- UNIX Vendors & Licensing Agreements
- AT&T UNIX Version 7 & System III & V
- Berkeley UNIX 4.2BSD ■ User Groups

Survey of UNIX Implementations
- AT&T ■ DEC ■ DG ■ HP ■ IBM-AT/RT
- Intel ■ Motorola ■ National ■ Sun ■ TEK
- Zilog ■ Apollo ■ Gould

9. FUTURE DEVELOPMENTS

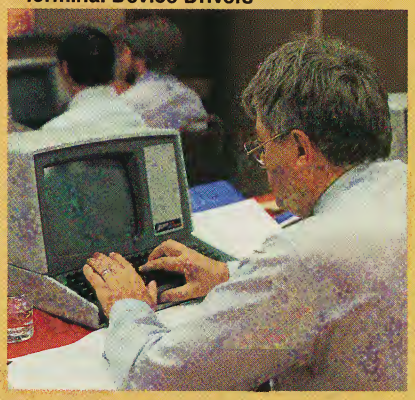
- Window-Oriented Shells & Editors
- Networks & "Distributed" UNIX Systems
- Growth in Appplications Software

HANDS-ON

This course emphasizes in-class, hands-on exercises using the UNIX operating system.

Hands-on exercises include:

- **Using Shellscripts in Applications**
- **Executing Programs under UNIX**
- **Employing UNIX Pipe & Filter Facilities**
- **Using Files, Directories & Devices**
- **Sending & Receiving Electronic Mail**
- **Terminal Device Drivers**



Course participants executing in-class exercises on a multi-user system.

INTEGRATED COMPUTER SYSTEMS

ENROLL NOW!

Call Toll Free **(800) 421-8166** or

East Coast Office **(703) 847-6448**

West Coast Office **(213) 417-8888**

Software Requirements & Specifications

8805 UB 311

Course 311

NEW!



3 Days

PALO ALTO
June 8-10
WASHINGTON, D.C.
June 29-July 1
SAN DIEGO
July 13-15
TORONTO
July 13-15
SAN FRANCISCO
August 10-12
WASHINGTON, D.C.
August 17-19
LOS ANGELES
September 14-16
BOSTON
September 21-23
WASHINGTON, D.C.
November 2-4

Courses at Your Facility
Please Inquire

Course Benefits

Focusing adequate attention and resources on the initial phases of the Software Development Life Cycle is the key to avoiding the usual software development crisis. Better understanding of end-user requirements together with complete system and software specifications, can prevent cost overruns, slipped schedules, and excessive maintenance.

This course provides participants with **specific guidelines and practical methods for effectively developing requirements and specifications, thereby ensuring correct, complete and reliable software systems.**

YOU WILL LEARN HOW TO:

- **Use Systematic Techniques for Analyzing and Documenting End-User Requirements**
- **Develop System and Software Requirements, Including Validation Criteria**
- **Develop Specifications for Software Design**
- **Incorporate the Unique Constraints of Real-Time Functions**
- **Adjust the Process for Systems of Differing Complexity**
- **Use a Wide Variety of Software Requirements & Design Tools**

WHO SHOULD ATTEND:

This course is valuable for software and system engineers, systems analysts and designers, programmers, product/project managers and technical marketing staff who are involved in specifying, developing or managing software projects. The course is valuable for those who are purchasing software systems as well as implementing them.

Course Content

1. INTRODUCTION AND OVERVIEW

The Software Development Life Cycle
- The Waterfall Model
- Baseline Management and Alternative Models
- Phases, Processes, Activities, Documents
- Sequential and Parallel Events
- Requirements Errors and Their Consequences
- Breakage: Causes and Effects

Key Development Issues
- Goals of Users, Purchasers, and Developers
- Change Control and Configuration Management
- Costing, Sizing and Scheduling
- Independent Verification and Validation

Software Development Standards
- Industry ■ Government
- Military (DoD-STD-2167)
- European and International Standards

Software Engineering Technology
- Structured Analysis and Design Techniques
- Object-Oriented Methodology
- Software Development Environments
- Re-using Software Modules
- "Rapid Prototyping" Goals and Tools
- CASE: Computer Assisted Software Engineering

2. APPLYING THE SYSTEM MODEL

The System Hierarchy
- System Elements and Levels
- Hardware, Software and Composite Elements
- The Documentation Tree
- Effects of Size and Complexity

Key System Documents
- The Concept-of-Operations: The "Why"
- Requirements Specifications: The "What"
- Design Specifications: The "How"
- System vs. Software Specifications
- Interface Specifications: Levels and Details
- The Build Plan and Test Documents

Allocation, Flowdown, and Traceability
- Top-Level Architectural Design
- Software/Hardware Trade-Offs
- Types of Allocation
- The Flowdown Process
- Traceability to the Concept-of-Operations, Requirements, Design, and Test Documents

3. SPECIFYING THE REQUIREMENTS

Classes of Requirements
- Understanding Requirements Issues
- Determining Functional Requirements
- Defining Performance Requirements
- Identifying Key System and Software Constraints

The Concept-of-Operations Document
- Defining the User Interface
- A Template for the Concept-of-Operations
- Relationship to User Documentation
- Relationship to Acceptance Test Plans

Generating Requirements Specifications
- Iterating with the Concept-of-Operations
- Incorporating Real-Time Requirements
- Allocating Functions to the End User
- Development and Test Planning

Requirements Development Tools and Methods
- Requirements Template/Check List
- Key Criteria for Evaluating Requirements
- Technical Writing Guidelines
- Analytical Tools: Structured Analysis, SADT, PSL/PSA, SREM, N² Charts
- Other Automated Tools
- Conducting Walkthroughs ■ Traceability
- Tools and Techniques for Handling Changes
- Prototyping and Simulation

4. THE DESIGN SPECIFICATION

Specification Processes and Products
- Distinguishing the "What" from the "How"
- Defining Programs and Modules
- The Transition from Requirements to Design
- Comparing Requirements and Design Specifications

Modularization Techniques
- Modularization and Decomposition
- Modularization Criteria and Evaluation
- Allocating Functions to Modules
- Planning for Module Testing

Specifying The Design
- Format and Contents
- Specifying Interfaces
- Data Structure Specifications
- Integrating and Updating Previous Products
- Designing to Cost
- Incorporating Test Planning

Design Specification Tools and Methods
- Functional Design: Structured Analysis, DFDs, Warnier-Orr Diagrams, PDLs
- Object-Oriented Design
- Validation: Walkthroughs, Design Reviews
- Automated Design Tools

5. PREPARING FOR CODING, TESTING, INTEGRATION AND MAINTENANCE

Principal Products
- Requirements and Design Specifications
- Interface Specifications
- Test Plans ■ Other Plans
- User Documentation
- Configuration Management Procedures

Phasing the Work Efforts
- Sequential Activities
- Parallel Activities
- Fallback and Contingency Planning

Staffing Considerations
- The Independent Test Team
- Engineering and Coding Activities
- Staff Training for Integration and Test
- Independent Verification and Validation

Course Workshops

Participants will work in teams to reinforce practical knowledge and skills needed for:

- **Developing Primary Systems Documents**
- **Allocating Requirements and Generating Software Design Specifications**
- **Evaluating the Traceability of Requirements, Design, and Test**
- **Requirements Evaluation**

INTEGRATED COMPUTER SYSTEMS

ENROLL NOW!

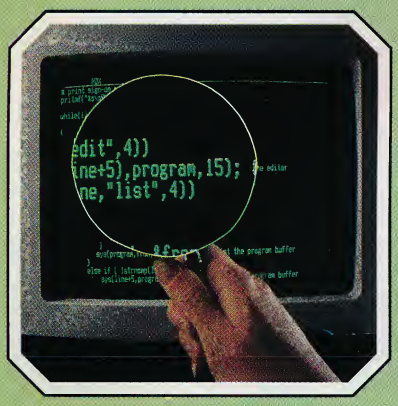
Call Toll Free **(800) 421-8166** or East Coast Office **(703) 847-6448** West Coast Office **(213) 417-888**

8805 UB 312

Software Quality Assurance & Testing

Course 312

NEW!



3 Days

WASHINGTON, D.C.
July 13-15

SAN DIEGO
July 20-22

TORONTO
July 20-22

SAN FRANCISCO
August 17-19

WASHINGTON, D.C.
August 24-26

LOS ANGELES
September 21-23

BOSTON
September 28-30

WASHINGTON, D.C.
November 16-18

Courses at Your Facility
Please Inquire

Course Benefits

The cost-effective production of quality software which meets user requirements is the goal of every software development project. To achieve this goal, rigorous Quality Assurance and Testing procedures must be applied at each phase of the Software Development Life Cycle.

This course provides participants with the **practical skills needed to implement effective QA and Testing** activities by providing procedures, checklists, and specific techniques for each phase of the life cycle. These techniques are reinforced through a series of in-class workshops which demonstrate practical applications of concepts as they are presented.

YOU WILL LEARN HOW TO:

- **Verify and Validate the Quality of the Products at Each Phase of the Software Development Life Cycle**
- **Design Test Scenarios and Data Sets to Insure Proper Testing**
- **Develop an Integration Test and Build Plan**
- **Select Appropriate Hardware and Software for Test Support**
- **Use Configuration Management to Support QA and Testing**
- **Validate and Deliver the Final Products to the User**
- **Determine Resources Required for Effective QA and Testing**

WHO SHOULD ATTEND:

This course is valuable for software quality assurance and testing specialists, software engineers, systems analysts, programmers, technical marketing personnel, auditors and project managers who are responsible for evaluating, developing, testing, or managing software projects of any size.

Course Content

1. INTRODUCTION AND OVERVIEW

The Role of Quality Assurance and Testing
- Defining QA, Testing, and Independent Verification and Validation (IV&V)
- The High Cost of Poor Quality Software
- Cost Benefits of QA and Testing
- Organizational Alternatives for QA, Testing and IV&V

Software Development Life Cycle (SDLC)
- Major Phases of the SDLC: Requirements, SW Design, Coding, Integration, Validation, Maintenance
- Specific Products of Each SDLC Phase
- Quality Assurance and Testing by Phase

The Concept of Incremental Development
- Benefits of Incremental Development
- Evolution and Iteration

2. INITIAL PLANNING FOR QA & TESTING

Choosing a Standard
- IEEE ■ FIPS-PUBS ■ MIL-specs
- International Software Standards
- Using Organizational Standards

Configuration Management: A Key QA Tool
- Goals and Objectives
- The Concept of Baselines
- Techniques for Controlling Change
- Adjusting to Changing Requirements
- Tracking Procedures for Action Items

Effective Techniques for Meetings
- Reviews for Small and Large Projects
- Guidelines for Conducting Walkthroughs
- Inspecting Life Cycle Products
- Key Life Cycle Milestone Audits

3. QA DURING THE REQUIREMENTS PHASE

The Concept-of-Operations Document
- Identifying the "Real" User
- Testing the Concept-of-Operations
- Planning for System Acceptance Testing

"Proving" System Requirements
- Traceability to the Concept-of-Operations
- Manual and Automated Requirements Checking Techniques
- Prototyping and System Simulation
- Checklist for Reviewing Requirements

4. QA DURING DESIGN AND CODING

Proving the Software Design
- Decomposition and Modularization
- Cost and Schedule Models
- Characteristics of "Good" Programs
- Planning for Program Validation Testing

Proving the Program Design
- Verifying the Code: Goals and Objectives
- Analyzing Structured Code
- Using Automated Static Analysis Tools
- Using Automated Dynamic Analysis Tools

Information Dissemination During Coding
- Programmer Development Folders
- Module Prologues and Comments
- QA of Code Walkthroughs
- Role of Test During Code Walkthroughs

5. MODULE-LEVEL QA TESTING PHASE

Module Testing Methodology
- Desk Checking and Code Inspections
- Designing Test Cases
- Generating Test Data
- Writing Program Stubs and Drivers
- Replacing Stubs and Drivers with Modules
- Black-box (Data Driven) vs. White-box (Logic Driven) Testing
- Simulating the Operating Environment
- Dealing With Real-time Constraints
- Techniques for Effective Debugging

Managing the Module Test Phase
- A Simple Method for Tracking the Error Correction and Module Re-test Cycle
- Key Error Collection and Analysis Techniques

Module Testing Tools and Support
- Stand-Alone Development Systems
- Host-Target Configurations
- Debugging With Interactive Tools
- Monitoring Test Coverage
- Test Matrices
- Test Data Generators

6. INTEGRATION PHASE QA AND TESTING

The Integration Test Strategy
- Dealing with Resource Limitations
- Choosing the Right Build Philosophy: Top-down? Bottom-up? Thread? Capability?
- Phased vs. Incremental Builds

Implementing the Test Plan
- Iterating Test and Build Functions
- Optimal Regression Testing
- Test Specification and Procedures
- Scheduling Tests and Reporting Results
- Anticipating Test and Rework Bottlenecks

7. SYSTEM VALIDATION AND ACCEPTANCE TESTING

Goals and Objectives of System Validation
- Validating that the Final System Build Matches the System Specification
- Simulating Hardware Failures
- Testing the System/User Interface
- Revalidating Revised Requirements
- Performance Under Maximum Load
- Pushing the Performance Envelope
- Validating User and Operations Manuals

System and Acceptance Testing
- Who Performs the System Test?
- Using Integration Test Results
- Test Completion Criteria and Delivery
- Validating Version and Release Controls
- The Software Maintenance Testing Plan

Course Workshops

During the course workshops participants will gain practical experience with QA and testing for each phase of the life cycle:

- **Estimating QA and Test Resources**
- **Evaluating a Requirements Document Using Structured Techniques**
- **Using a Test Matrix Technique to Develop Test Methodology Plans**
- **Conducting a Test Case Development Walkthrough**

INTEGRATED COMPUTER SYSTEMS

ENROLL NOW!

Call Toll Free **(800) 421-8166** or

East Coast Office **(703) 847-6448**

West Coast Office **(213) 417-8888**

Software Maintenance

8805 UB 314

Course 314 **NEW!**



3 Days

LOS ANGELES
August 31-September 2
WASHINGTON, D.C.
September 14-16
BOSTON
October 5-7
MONTREAL
November 2-4
TORONTO
November 30-December 2
WASHINGTON, D.C.
November 30-December 2
SAN DIEGO
December 14-16
SAN FRANCISCO
January 25-27

Courses at Your Facility
Please Inquire

Course Benefits

Software Maintenance typically accounts for 50% to 80% of the entire software budget. This course presents systematic tools and techniques which can greatly reduce maintenance costs, increase programmer productivity and improve user satisfaction. These benefits apply both to older, unstructured code as well as to software produced with modern techniques.

Participants are provided with the knowledge and skills required to effectively plan, implement and control the software maintenance function. The course emphasizes pragmatic guidelines and specific practices for use within both maintenance and development organizations.

YOU WILL LEARN HOW TO:

- **Organize, Plan, Monitor, and Control Software Maintenance**
- **Define the Tools and Environment Required for Maintenance**
- **Apply Software Engineering and Structured Techniques for Modifying and Maintaining Software**
- **Re-Engineer Unstructured Programs to Improve Maintainability**
- **Implement Software Development Procedures to Reduce Future Maintenance Costs of Software Now Under Development**
- **Develops Life Cycle Maintenance Strategies**

WHO SHOULD ATTEND

This course is valuable for systems and software maintenance personnel, software engineers, programmers, managers, and other technical professionals and users who wish to improve their techniques and systems for software maintenance. Familiarity with software development methods is useful.

Course Content

1. INTRODUCTION AND OVERVIEW

Understanding Software Maintenance
- Defining the Maintenance Function
- Establishing Goals and Priorities
- Classifying Maintenance Activities
- Estimating the Real Cost of Maintenance

Software Maintenance Relationships
- Integrating Maintenance into the Software Development Life Cycle (SDLC)
- Interacting with HW/SW System Activities
- Impact of Application Type: Database, Real-time, Military, Commercial, Others

Software Maintenance Management
- Organizing and Planning for Maintenance
- Providing Tools for Maintenance Support
- Implementing & Controlling the Process

2. SOFTWARE MAINTENANCE ENGINEERING

The Maintenance Phase
- Using Products of each SDLC Phase
- The User's Responsibilities
- Structuring the Maintenance Process

Defining the Requirements for Change
- Identifying Sources of Change Requests
- Analyzing the Needs of the User
- Classifying Change by Type: Corrective, Adaptive, Perfective

Implementing the Change
- Investigating & Evaluating the Request
- Understanding the Software & Its Functions
- Analyzing "Ripple" Effects of the Change
- Designing & Implementing Correct Solutions
- Documenting All of the Changes

Testing & Assuring Modified Software
- Critical Importance of Testing
- Tracing Changes in Requirements, Specifications & Code
- Using Simulation Techniques
- Regression Testing & Re-Validation
- Accepting the Revised Software
- Reviews & Audits

Improving Existing Software
- Measuring Source Code Maintainability
- Using Reverse Engineering Techniques
- When to Rewrite Software Products

Managing the Software Configuration
- Configuration Management Activities
- Tracking, Auditing, & Status Reporting
- Establishing Effective Change Control
- Scheduling the Release of Revised Products
- Controlling/Supporting Multiple Releases

3. PLANNING FOR MAINTENANCE IN THE SOFTWARE DEVELOPMENT CYCLE

Designing for Maintainability
- Maintainability Characteristics of SW
- SW Quality Attributes & Maintenance
- Measuring Maintainability Using Quality and Status Metrics
- Using Standards to Ensure Consistency
- Development & Maintenance Documentation
- The Development Environment
- Software Assurance & User Acceptance

Developing Maintainable Software
- How to Plan for Maintenance
- Using Software Engineering Techniques that Insure Maintainability
- Language Features & Maintainability
- What the Maintenance Engineer Really Needs to Know About the Software

4. SW MAINTENANCE SUPPORT ENVIRONMENTS

Using Technical Maintenance Tools
- CASE Tools & Workbenches
- Test Case Generators and Test Drivers
- Special Debugging and Maintenance Tools
- Applying Tools for Assurance & Analysis

Support Tools and Product Information
- Configuration Management Tools
- Tools for Documenting and Communicating
- Special Tools for Supporting Management
- Accessing SDLC Products & Release Documents
- Building the Maintenance Library Facility

5. MANAGING SOFTWARE MAINTENANCE

Management Activities
- Planning and Coordinating the Process
- Controlling the Rate of Changes
- Maintaining Software Quality
- Using Status and Management Metrics
- Defining, Selecting, Educating, and Motivating Maintenance Personnel

Organizing the Maintenance Team
- Coordinating Development, Operations & Maintenance
- Defining Task Responsibilities
- When to Cross Train the Software Team

Maintenance as a System Function
- The Functions of Maintenance and Development in the System Life Cycle
- Checklist of System & SW Documentation
- Interaction of SW & System Changes

6. FUTURE DIRECTIONS IN SW MAINTENANCE

- Prototyping Impacts on the Life Cycle
- Tailoring the Life Cycle to Accommodate Phased Deliveries
- Evolutionary Development and Spiral Life Cycles
- Using Fourth Generation Languages
- Impacts of Reusable Code

Course Workshops

During the course participants develop systematic techniques for solving maintenance problems by applying five key concepts to real-world problems.

The skills gained by participants in these workshops can be used immediately upon their return to work.

- **Evaluate Software for Maintainability Using Checklists and Quality Metrics**
- **Select Appropriate Solutions for an Emergency "Patch"**
- **Modify the Implementation of a Current Capability**
- **Enhance a System by Developing a New Capability**
- **Analyze the Software Impact of a Requirements Change and Define Solution Alternatives**

INTEGRATED COMPUTER SYSTEMS

ENROLL NOW!

Call Toll Free
(800) 421-8166 or

East Coast Office
(703) 847-6448

West Coast Office
(213) 417-8888

8805 UB 306

Software/User Interfaces: Design Principles & Techniques

Course 306 DEMO LABS



4 Days

LOS ANGELES
June 14-17
WASHINGTON, D.C.
June 21-24
WASHINGTON, D.C.
September 13-16
TORONTO
September 27-30
LOS ANGELES
November 1-4
WASHINGTON, D.C.
November 29-December 2
SAN DIEGO
December 6-9

Courses at Your Facility
Please Inquire

Course Benefits

Dramatic improvements in Software/User Interfaces are being fueled by the rising expectations and demands of computer users for "easy to learn **AND** easy to use" software, combined with the widespread availability of powerful PC/Workstations with large memories and high resolution displays.

This course provides participants with the fundamental techniques and practical tools for the implementation of effective Software/User Interfaces. In-Class Demonstrations and case studies ensure that participants understand, and are able to apply, the software design and development methods presented.

YOU WILL LEARN HOW TO:

- **Understand the Key Design Techniques of User/SW Interfacing**
- **Design Interfaces that are Easy to Learn AND Easy to Use**
- **Develop Effective Screens, Windows, Dialogs, and Menus**
- **Design Mouse-Driven/High Resolution Display Interfaces**
- **Employ Off-the-Shelf Prototyping Tools and SW Components**
- **Improve User Productivity & Product Acceptance**

WHO SHOULD ATTEND:

This course is valuable for programmers, systems analysts, software engineers, and technical managers who are responsible for the specification, system design or software development of modern Software/User Interfaces.

Course Content

1. INTRODUCTION AND OVERVIEW

Fundamental Concepts and Terminology
- What is a "Superior" User Interface
- Viewing the Software/User Interface as a Communication Process
- Designing for the User: Achieving "Ease of Use" & "Ease of Learning"

The Evolution of Modern User Interfaces
- Command vs. Menu-Driven Interfaces
- High Resolution, Bit-Mapped Displays
- Pointing Devices: Mouse, Trackball, Tablet
- Windowing Environments & Icons
- Dialog Boxes & "Forms"
- Emergence of Voice Input/Output
- Impact and Evolution of "Standards"

Benefits of Effective User Interfaces
- Improved Product Acceptance
- Increased User Productivity
- Reduced Training Time & Cost

Applications Overview
- Office Automation ■ Information Systems
- Mini Computer/Multi User Databases
- CAD/CAM/CAE ■ Military/Aerospace
- PC/Workstation Software Products

2. FUNDAMENTALS OF SOFTWARE/USER INTERFACES

Steps in Designing Human-Computer Interfaces
- Defining Performance Objectives
- Evaluating HW, SW, and Human Components
- Designing the Software/User Interfaces
- Iterative Test & Optimization

Human Factors in User Interfaces
- Understanding & Designing for the User
- Feedback and the Learning Process
- Predicting Errors ■ Avoiding Confusion
- Perceived vs. Actual Performance

Key Software Design Considerations
- Identifying System Constraints
- Layered Design, Modularity
- Designing for Device Independence

Program Development Tools
- Software Development Environments
- Languages ■ Rapid Prototyping Tools
- Simulation Aids ■ Module Libraries
- Database & Expert System Front Ends

3. DESIGNING EFFECTIVE SCREENS & MENUS

Screen Layout & Design
- Tools for Prototyping Screen Layout
- Window Positioning: Tiled vs. Overlapped
- Use of Bold, Blinking, Reverse Video, Color
- Proper Use of Icons
- Common Mistakes in Screen Design

Menu Design
- Developing the Menu Layout & Tree Structure
- Menu Placement and Persistence
- Pop-Up, Drop-Down & Fixed Menus
- User Orientation & Menu Navigation
- Item Selection Techniques

4. DEVELOPING EFFECTIVE INTERACTION

Providing the User with Status Information
- Getting the User's Attention
- "Current Status" Information
- Waiting for Input vs. "Working"
- Options/Guidance for Next User Action

Dialogue Boxes and Form Filling
- Specifying Parameters of an Operation
- Buttons, Lists, Text Input, Check Boxes
- Proper Use of Defaults

Improving the Speed of Interaction
- User "Self-Customization" of Interaction
- Keystroke Sequences in Lieu of Pointing
- Verbose and Terse Modes of Interaction
- User Defined Menus & Keyboard Macros
- "Undo" & "Redo" Facilities

Designing Effective Help Systems
- Design of Help Screens
- Context Sensitive Help
- Multiple Level & "User Adjustable" Help
- Providing Help after Errors
- Help as a Tool for Training Users

5. IMPLEMENTING WINDOW APPLICATIONS

Windowing Environments
- Understanding Windowing Techniques
- Using Window Tool Kit Components
- Font Editors ■ Icon Generators
- Using the Virtual Device Interface (VDI)

Survey of Windowing Packages
- Key Differences Between Packages
- Apple/MacIntosh ■ IBM/Topview
- MS/Windows ■ SUN/Windows
- XWindows ■ Apollo

Implementation Guidelines
- Application Analysis & Design
- Designing for Portability
- Dealing With Multiple Environments
- Porting Existing Applications to Windows

6. APPLICATION CASE STUDIES

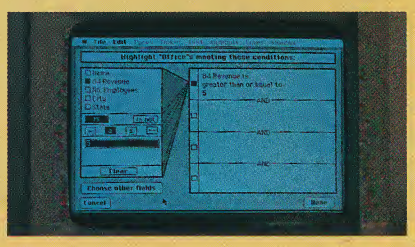
- Mouse-Driven High Resolution
- Computer Aided Design System
- Multi-Level Database Interface
- Business Graphics System
- Developing an On-Line Tutorial

DEMO LABS

Participants will have the opportunity to see key SW/User Interface concepts, techniques and tools demonstrated throughout the course on an In-Class Workstation.

Case studies and examples include:

- **Using Rapid Prototyping Tools**
- **Creating Screens, Forms and Menus**
- **Editing Icons and Fonts**
- **Using Window Environments & Tools**



INTEGRATED COMPUTER SYSTEMS

ENROLL NOW!

Call Toll Free **(800) 421-8166** or

East Coast Office **(703) 847-6448**

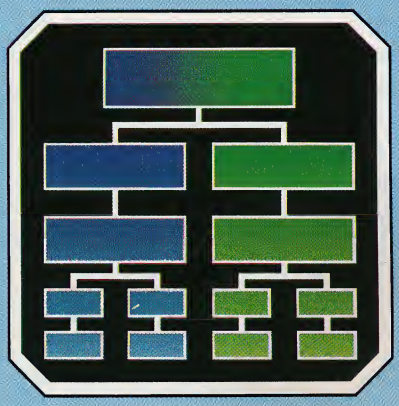
West Coast Office **(213) 417-8888**

UB

Structured Design & Programming

8805 UB 320

Course 320



4 Days

BOSTON
May 24-27
LOS ANGELES
June 14-17
TORONTO
June 14-17
WASHINGTON, D.C.
June 21-24
LOS ANGELES
August 30-September 2
TORONTO
September 13-16
WASHINGTON, D.C.
September 27-30
SAN DIEGO
October 18-21
OTTAWA
October 25-28
BOSTON
November 1-4

Courses at Your Facility
Please Inquire

Course Benefits

The use of structured programming methods and tools has been proven successful in **significantly reducing program errors, increasing productivity and improving the long-term maintainability of software.** This unique course presents practical, systematic techniques for structured design and coding in real-time, engineering and scientific applications.

The course is a highly interactive workshop providing both the fundamentals and practical applications of structured programming methods. The presentation incorporates a series of case studies in which participants develop skills for efficiently producing reliable, well-documented, and maintainable programs — on time and within budget.

YOU WILL LEARN HOW TO:

- **Write Programs That Execute Correctly the First Time**
- **Utilize Structured Software Design Methods for Engineering and Scientific Applications**
- **Increase Programming Productivity**
- **Improve Program Readability and Reliability**
- **Detect Logic and Coding Errors Early**
- **Insure Effective Testing and Documentation**
- **Reduce Software Development and Maintenance Costs**

WHO SHOULD ATTEND:

This course is valuable for software and hardware engineers, applications and systems programmers, systems analysts, designers, project leaders and others who are active in software development projects.

Course Content

1. INTRODUCTION & OVERVIEW

Characteristics of Bad Programs
- "Holy" Programs ■ Waterfall Code
- Tricky Code ■ Diffuse Logic
- Mystery Names ■ Dead Code

Characteristics of Better Programs
- Clearly Delineated Control Structure
- Modularity ■ Limited Complexity
- Descriptive Names ■ Readability
- Reliability ■ Ease of Testing
- Decreased Maintenance Costs

2. BASIC DESIGN PRINCIPLES

Fundamental Expressions & Structures
- Hierarchical Design & Construction
- Using Pseudo-Code
- Program Development Language (PDL)
- Sequence-Selection-Iteration
- IF-THEN-ELSE ■ Cascading IFs
- WHILE-DO vs. DO-UNTIL vs. FOR
- CASE ■ Table-Driven Logic

Unit Design Tools and Techniques
- Stepwise Refinement
- Backus-Naur Form (BNF)
- Sets and Set Notation
- Warnier-Orr Diagrams

Program Complexity
- Human Capacity for Complexity
- Sources of Program Complexity
- Control Flow ■ Size
- Interrelations ■ Incongruence
- Measures of Complexity

Basic Testing Concepts
- Function Testing ■ Path Testing
- Static vs. Dynamic Testing
- Automated Testing ■ Debugging

3. SOFTWARE ENGINEERING PRINCIPLES

Focus of Software Engineering
- Resolving Hardware vs. Software Cost
- Coping with Complexity and Uncertainty
- Facilitating Change & Maintenance
- Avoiding Costly, Early Errors

The Dynamic Product Life Cycle
- Iterative Analysis and Synthesis
- Defining Levels of Abstraction
- Prototyping vs. Rigid Life Cycle

Insuring Software Quality
- Futility of Exhaustive Testing
- Fundamental Engineering Procedures
- Validation & Verification Steps
- Reviews & Inspections
- Progress Reporting & Summaries

Building MIL-STD Systems
- MIL-STD-1521 & 1679
- MIL-STD-483 & 490
- DoD Standard 2167

4. A DATA VIEW OF THE SYSTEM

Logical Analysis & Design Tools
- Goals for Logical Design
- Data Flow Diagrams (DFDs) ■ Bubbles
- Data Stores ■ Sources/Sinks
- Context, Overview and Detail DFDs
- Validating Data Flows ■ Leveling

Defining Data Structure
- Terminology ■ Bachman Diagrams
- Lists ■ Stacks ■ Queues ■ Trees
- Detailed Data Definition

5. PHYSICAL DESIGN

Goals of Physical Design
- Portability ■ Reuseability
- Reliability ■ Simplicity
- Efficiency vs. Modularity

Deriving Alternate Physical Designs
- Data Stream Analysis (Factoring)
- Afferent/Efferent Guidelines
- Asynchronous Processes
- Geographic Distribution

Choosing a Design to Optimize Goals
- Prioritizing Goals
- Tradeoff Analysis
- Completing the Design

6. DESIGNING CONGRUENT STRUCTURES

Design Based on Data Structure
- Processing Logic Structures
- List ■ Stacks ■ Queues
- One-to-one Mappings

Data Incongruence
- Logical vs. Physical Structures
- Boundry Clash ■ Structure Clash
- Synchronizing Events ■ Handling Errors
- Hierarchy & Program Inversion
- Structure Resolvers ■ Coroutines

Design Based on Process Structure
- Time Cycles ■ Monitors ■ Games
- Simulators ■ Command Processors
- Man-Machine Dialogs ■ Graphics
- Parameterized Programs ■ FSMs
- Tables, Rules & Engines

7. MODULAR PROGRAM DESIGN

Object-Oriented Design
- Achieving Software Engineering Goals
- Process Abstractions ■ Secrets
- Information Hiding ■ Data Abstractions

Defining Good Modules
- Attributes of Modules
- Measuring Module Goodness
- Cohesion ■ Coupling
- Pathological Connections

Other Design Heuristics
- Decision-Splitting
- Control vs. Transformation
- The "Mosque-Shaped" System

8. CONVERTING THE DESIGN TO CODE

Comparison of Languages
- BASIC ■ FORTRAN ■ COBOL
- Ada ■ Pascal ■ C ■ RATFOR
- Assemblers ■ Modula 2

Language and Program Style
- Block vs. Modular Structure
- Structuring for Efficiency
- Fourth Generation Procedural Languages
- Prototyping & HOL's

9. REAL-TIME SYSTEM DESIGN

Real-Time Considerations
- Efficiency vs. Modularity
- Concurrency ■ Deadlock
- Interrupt Processors ■ Monitors

Operating System Components
- Dispatchers ■ Schedulers
- The Ready Queue
- Avoiding Contention
- Messages & Rendezvous

10. CONCLUSION

- Design Principles
- A Coherent Development Strategy

INTEGRATED COMPUTER SYSTEMS

ENROLL NOW!

Call Toll Free **(800) 421-8166** or

East Coast Office **(703) 847-6448**

West Coast Office **(213) 417-8888**

CASE: Computer-Aided Software Engineering, Hands-On

HANDS-ON

Course 324



4 Days

BOSTON
June 7-10
OTTAWA
June 14-17
WASHINGTON, D.C.
August 2-5
WASHINGTON, D.C.
October 25-28
SAN FRANCISCO
November 15-18
TORONTO
November 22-25
BOSTON
December 6-9
WASHINGTON, D.C.
January 24-27

Courses at Your Facility
Please Inquire

8805 UB 324

Course Benefits

A new generation of Computer-Aided Analysis and Design Tools are having a profound and growing impact throughout the software life cycle. These tools enhance productivity and software quality while drastically reducing the cost of system development, maintenance and documentation.

The best way to learn how to use these tools is to **gain Hands-On Experience under the guidance of an expert instructor.** This course provides participants with the background and practical insights required to evaluate, select and utilize Computer-Aided Software Engineering Tools through In-Class Hands-On Exercises, Demonstrations and Case Studies.

YOU WILL LEARN HOW TO:

- **Produce Data Flow Diagrams, Hierarchy Charts & Data Dictionaries Using High-Resolution Graphical Design Aids**
- **Employ Rapid Prototyping Tools to Speed-Up Design**
- **Use Tools for the Prevention & Early Detection of Errors**
- **Utilize Integrated Text/Graphics Documentation Tools to Provide Project Visibility & Management Control**
- **Improve Productivity & Program Quality Using CASE Tools**

WHO SHOULD ATTEND:

This course is valuable for system analysts, designers, programmers, engineers and managers who are responsible for specifying, designing, developing or maintaining reliable software systems and applications. Prior exposure to structured techniques will be beneficial.

Course Content

1. INTRODUCTION & OVERVIEW

Basic Concepts & Terminology
- The Software Development Life Cycle
- Software Engineering & Structured Methods
- Evolution of Automated Design Tools

Computer-Aided SW Engineering Tools
- Graphical Tools for Analysis & Design
- Data Dictionaries & Project Libraries
- Rapid Prototyping Tools & Techniques
- Error Detection & Consistency Checking
- Code Generators & "Software Bridges"
- Documentation & Reporting Tools
- Tools for Project Management & Control
- Integrated Software Life Cycle Packages

Benefits of Computer-Aided Tools
- Increased Productivity
- Improved Quality & Performance
- Prevention/Early Detection of Errors
- Reduced Testing & Maintenance Costs
- Management Visibility & Control

2. HANDS-ON EXPERIENCE WITH ANALYSIS AND DESIGN TOOLS

Using the In-Class Workstation
- Understanding the User Interface
- Using a Mouse with Graphical Modeling Tools
- Excelerator: An Integrated Workbench

Developing an Initial System Design
- Creating a Data-Flow Diagram (DFD)
- Generating State Transition Diagrams
- Utilizing the Data Dictionary
- Checking for Errors & Inconsistencies

3. USING GRAPHICAL TOOLS FOR STRUCTURED SYSTEM DESIGN

Tools for Logical System Design
- Creating Multi-Level Data Flow Diagrams
- Structure Charts & Logical Data Models
- Entity-Relationship Diagrams
- Real-Time State Transition Diagrams
- Translating between Models

Program Design Support
- Warnier-Orr Charts ■ HIPO Diagrams
- Object-Oriented Methods
- Module Charts for Scoping Variables
- Use of Pseudo-Code Generators

Using Rapid Prototyping Tools
- Screen Generation & Painting
- Simulating User/System Dialog
- Using Report Generators

4. SUPPORT TOOLS FOR ANALYSIS, DOCUMENTATION & MANAGEMENT

Data Dictionaries & Project Libraries
- Data Elements ■ Process Names & Specs
- Maintaining Relationships Between Entities
- Change Control & History ■ Audit Trails

Error Prevention & Detection
- Checking for Consistency & Completeness
- Spotting Inconsistent/Missing Information
- DFD Level-to-Level Balancing
- Automated Rule Enforcement
- Tracing & Validating Requirements

Tools for System Documentation
- Using Integrated Graphics/Text Tools
- Developing Cross Referencing Indexes
- Reporting Analysis & Design Details
- Creating a Final System Specification

Project Management Tools
- Project Planning & Estimating
- Task Assignment & Release
- Configuration Management Reports
- Coordinating the Project Team

5. SELECTION & IMPLEMENTATION

Comparison Criteria
- Scope of Tool Coverage & Applications
- Knowledge/Rules Content
- Tool Useability

Comparison of Tools
- Integrated Engineering Workbench (KnowledgeWare)
- Analyst & Designer Toolkit (Yourdon)
- DesignAid & CASE 2000 (NASTEC)
- Excelerator (Intech) ■ PCSA (CADRE)
- ProMod ■ ER-Designer (Chen)
- Structured Architect & PSL/PSA (ISDOS)
- Structure(s) (Orr) ■ Teamwork/SA (Cadre)
- Visible Analyst (Visible Systems)
- I.D.E. ■ ANATOOL ■ PRISIM

Implementation Considerations
- Life Cycle Packages vs. Individual Tools
- SW Bridges to Code Generators & 4GLs
- Export of Data to Other Tools
- Interfaces to Mainframe Data Dictionaries
- Implementing a Pilot Project
- Training Requirements
- Strategic Considerations/Pitfalls to Avoid

6. ADVANCED TOPICS & FUTURE TRENDS

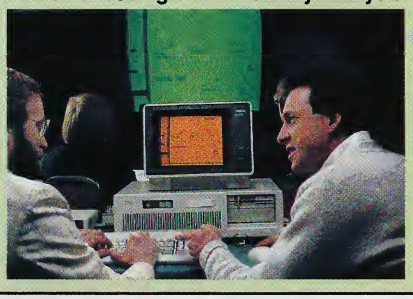
- Expected Tool Enhancements
- Evolution of Networking Environments
- Role of Knowledge-Based Systems

HANDS-ON

Throughout the course, participants gain Hands-On experience with a variety of PC-Based Software Analysis & Design Tools.

Hands-On exercises include:

- **Using Data Flow Diagrams to Develop a Top-Down System Design**
- **Creating a Data/Design Dictionary**
- **Defining Modules with Structure Charts**
- **Performing a Weighted Trade-Off Analysis**
- **Error Checking & Consistency Analysis**



INTEGRATED COMPUTER SYSTEMS

ENROLL NOW!

Call Toll Free **(800) 421-8166** or

East Coast Office **(703) 847-6448**

West Coast Office **(213) 417-8888**

Computer Graphics: Software Techniques & Hardware Components

8805 UB 365

DEMO LABS

Course 365



4 Days

LOS ANGELES
June 28-July 1
WASHINGTON, D.C.
July 19-22
OTTAWA
July 26-29
BOSTON
September 20-23
SAN FRANCISCO
October 11-14
WASHINGTON, D.C.
November 1-4
TORONTO
November 15-18
LOS ANGELES
November 29-December 2

Courses at Your Facility
Please Inquire

Course Benefits

Computer graphics has become a standard, cost-effective tool for virtually every area of engineering, business, and science. Rapid advances in the technology of powerful engineering workstations, special purpose graphics chips, intelligent terminals and personal computers have been combined with improvements in the quality and availability of device independent graphics software, user interfaces and software tools for application development.

This course provides a comprehensive overview of the State-of-the-Art in Computer Graphics software and hardware, and presents an integrated approach to the implementation of graphics applications.

YOU WILL LEARN HOW TO:

- **Apply the Fundamental Concepts of Computer Graphics**
- **Use Raster, Vector and Color Techniques**
- **Implement 2D and 3D Graphics Software Algorithms**
- **Make Use of Recent Graphics Standards**
- **Evaluate State-of-the-Art Graphics Workstations, Terminals and Plotters**
- **Compare the Performance of Graphics Software Packages**
- **Integrate Software and Hardware to Implement a Variety of Graphics Applications**

WHO SHOULD ATTEND:

This course is intended for engineers, scientists, systems analysts, programmers, and managers who are involved in specifying, implementing or using computer graphics systems.

Course Content

1. INTRODUCTION & FUNDAMENTALS

How Can Computer Graphics Help?
- Visualizing Complex Data Structures
- Computer Aided Design (CAD)
- Cost-Effective Simulation & Training
- Graphics Tools for Decision Making

Computer Graphics Systems Elements
- Graphics Display Hardware
- Interactive Graphics Input Devices
- Software Algorithms & Packages
- Applications Overview

Properties of the Visual System
- Illumination, Brightness & Contrast
- Resolution and Visual Acuity
- Color Perception

GKS & Other Key Graphics Standards
- The Benefits of Standards for Graphics
- Graphics Kernel System (GKS)
- Computer Graphics Interface (CGI)
- Computer Graphics Metafile (CGM)
- Others: PHIGS, GKS-3D, IGES

2. GRAPHICS HARDWARE

Input Devices for Graphics
- Digitizing Tablets ■ Joy Sticks ■ Track Balls
- The Mouse ■ Touch Panels ■ Light Pens

Display & Plotting Hardware
- Raster & Calligraphic Devices
- Dynamic Displays ■ Frame Buffers
- Color & High Resolution Devices
- Laser & Electrostatic Plotters

Graphics System Architectures
- Low Cost, Terminal-Oriented Systems
- Engineering Workstations and PC's
- Graphics Boards and Chips
- Dynamic, High Performance Systems

3. SOFTWARE FOR 2D GRAPHICS

Basic Algorithms
- Equations for Lines & Planes
- Coordinate Systems
- Vector & Matrix Operations
- Viewing & Object Specification

Interactive Graphics Techniques
- The Logical vs. Physical Input
- Creating User Dialogues
- Menus, Windows and Icons
- Rubber Banding, Relaxation, Dragging

Two-Dimensional Graphics & GKS
- Device Independent Software
- Windows & Viewports
- Segmentation and Transforms
- Attributes and Bundling
- Dynamic Screen Control
- Storage and Metafiles

Implementation
- The Graphics Pipeline
- Clipping Algorithms
- Primitive Generation
- Device Drivers and the CGI

4. THREE DIMENSIONAL GRAPHICS

- 3D Coordinate Systems
- Perspective & Parallel Projections
- Software: CORE, GKS-3D
- Implementation Pipeline

Modeling
- Hierarchical Models: PHIGS
- Polygonal Meshes
- Polynomial Curves & Surfaces
- Solid Models

Hidden Surface Removal
- Z-Buffer & Scan Line Algorithms
- Depth Sort Algorithms
- Area Subdivision Techniques

Rendering
- Color Models ■ Shading
- Shadows ■ Texture ■ Fractals
- Transparency ■ Reflection

5. RASTER GRAPHICS

Raster Graphics Techniques
- Pixels & Frames Buffers
- Rasterization & Scan Conversion
- Raster Ops
- Polygon Filling & Clipping Algorithms
- Antialiasing

Using Raster Scan Graphics
- Using Video Look-up Tables
- True, False & Pseudo Color
- Color Models for Raster Graphics
- Halftones and Dithering

6. IMPLEMENTING APPLICATIONS

System Design Considerations
- Requirements Specification
- Hardware/Software Tradeoffs
- Hidden Costs & Pitfalls to Avoid
- Impact of Evolving Standards

Available Equipment
- Turn-Key CAD Systems
- Terminal Based Systems
- Personal Computer Architectures
- A High Resolution Workstation
- Simulation/Image Generation
- Off-the-Shelf SW Components & Tool Kits
- Survey of Graphics Workstations: IBM RT, SUN-III, Vaxstation II/GPX, Apollo
- Graphics Boards for PCs & Workstations
- High-Speed Graphics Chip-Sets

DEMO LABS



The graphics techniques and applications presented in the course are illustrated by extensive use of computer-generated slides and video tapes.

Free Take-Home Software

Each participant will be provided with listings of computer graphics programs that illustrate specific implementations of the concepts presented in class.

The program listings consist of a set of device-independent modules, which can be used as models in the development of systems for the attendees' applications.

INTEGRATED COMPUTER SYSTEMS

ENROLL NOW!

Call Toll Free **(800) 421-8166** or

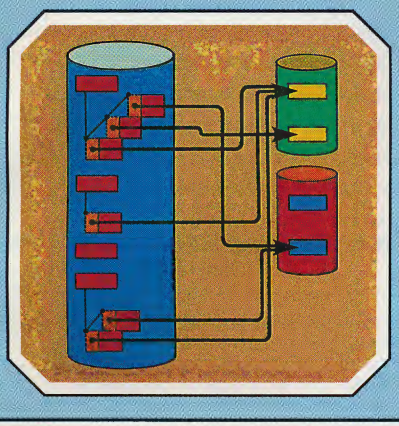
East Coast Office **(703) 847-6448**

West Coast Office **(213) 417-8888**

Relational Databases: Design, Tools & Techniques

DEMO LABS

Course 382



4 Days

OTTAWA
June 14-17
WASHINGTON, D.C.
July 12-15
LOS ANGELES
July 19-22
OTTAWA
July 26-29
SAN FRANCISCO
September 13-16
MONTREAL
September 20-23
WASHINGTON, D.C.
September 27-30
LOS ANGELES
December 13-16
WASHINGTON, D.C.
December 13-16

Courses at Your Facility
Please Inquire

8805 UB 382

Course Benefits

Relational Database Management Systems (DBMS) permit rapid implementation of sophisticated applications with minimal program development and maintenance cost. Relational DBMSs are now widely available for PCs and workstations, as well as minis and mainframes. They are being used successfully in a growing number of engineering, commercial and scientific applications.

This course provides participants with a comprehensive introduction to DBMSs. It provides a practical foundation in DBMS technology which will enable participants to understand, evaluate and effectively use commercially available relational DBMS packages.

YOU WILL LEARN:

- **How Relational Database Management Systems Work**
- **Criteria for Determining When a DBMS Is the Right Approach**
- **Methodologies for Designing a Relational Database**
- **Techniques for Analyzing Key Requirements**
- **How to Develop the Conceptual Data Model, and Map the Conceptual Model into the Relational Model**
- **How to Evaluate and Select a Relational DBMS Package**

WHO SHOULD ATTEND:

This course is valuable for programmers, systems analysts, engineers, scientists, planners and managers involved in implementing DBMS applications. Some background in programming fundamentals is recommended.

Course Content

1. INTRODUCTION

An Overview of DBMS Technology
- Key Concepts and Terminology
- DBMSs vs. Simple File Managers
- Data Independence and Structuring
- Single vs. Multi-User Environments

Requirements-Driven Database Definition
- Conceptual Schemas: The Data Model
- External Schemas: A User's View of Data
- Internal Schemas: Physical Data Structure
- Defining the Integrity Constraints

DBMS Applications on Micros and Minis
- Computer-Aided Engineering Tools
- Management Information Systems
- Transaction Processing ■ Manufacturing
- Research Databases
- On-Line Query Languages and Menus

2. HOW A RELATIONAL DBMS WORKS

Relational Database Fundamentals
- The Structure of a Relational Database
- Defining Attributes, Tables and Indices
- Relational Algebra: Selection, Projection, Join, Difference, Union
- Relational Integrity Rules
- 1-to-N and N-to-P Relationships
- Primary Indexing and Bit Indexing
- Secondary Indexing & Complex Networks
- Pointer Arrays ■ Implicit Linking

Using Query Languages
- Implementations of Relational Algebra
- SQL: A Structured Query Language
- QBE: A Syntax-Free Query Language
- CLOUT: A "Natural" Query Language
- Application Generators and Report Writers

Maintaining and Refining the Database
- Dynamic Manipulation of Database Content
- Adding New Attributes, Indices and Tables

Multi-User System Considerations
- Database Integrity and Security
- Locking to Provide Access Protection
- Resolving Deadlocks with Dynamic Backout
- Facilities for Importing and Exporting Data

Comparison with Hierarchical, Network/CODASYL
- Non-Procedural vs. Navigational
- Processing Multiple Records
- Join Relations vs. File Linking
- The Concept of Essentiality
- Ease of Database Evolution

3. BUILDING AN APPLICATION DATABASE

Key Relational Implementation Steps
- Developing Functional Specifications
- Defining Relations and Attributes
- Describing Interactive User Dialogs
- Specifying Screen and Report Layouts
- Using the Scenarios & Dictionaries Method

Entity Relationships Modeling
- Top-down Approaches
- Identifying the Entities and Rules
- Entity Relationships: One-to-One, One-to-Many, Many-to-Many

Logical Data Design
- Mapping to the Relational Model
- Designing User Views
- Normalizing the Data
- Building a Data Dictionary

Physical Data Design
- Fundamentals of Disk Structures
- Primary vs. Secondary Accesses
- Implementing Relationships
- Data Clustering and Compression

Program Design
- The Driving Role of Transactions
- Implementing Enterprise Rules: Syntactic, Semantic, Referential

Implementation Tools
- Language Interfaces: FORTRAN, PASCAL, C, COBOL, BASIC, Pre-Compilers
- Fourth Generation Languages
- Data Modeling Tools
- The Programmer's Workbench Concept

4. EVALUATING DBMS PACKAGES

Software Evaluation Considerations
- Specifying Application Requirements
- Performance ■ Documentation
- Relational Completeness
- Availability, Compatibility and Portability on Workstations, Minis and Mainframes

System-Related Considerations
- CPU Power and System Throughput
- Memory and Disk Capacity
- System Software Support

Evaluating Relational Systems
- Experimental vs. Actual Systems
- Key Performance Considerations
- Benchmarking Relational DBMS Packages

Review of Selected Packages
- Oracle ■ Ingres ■ R:BASE
- dBASE III Plus ■ Paradox ■ DB2
- FOCUS ■ Others

5. CASE STUDIES

Application Examples
- Databases for Computer-Aided Design
- Knowledge and Manufacturing Databases

6. FUTURE TRENDS

- Standardization of a Relational Interface
- Distributed Databases
- Database Machines
- Support for End User Computing

DEMO LABS

This course emphasizes the principles and practical applications of Relational Database Systems. **Key concepts and techniques are demonstrated on an in-class computer.** Case studies and examples include:

- **Database Design and Normalization**
- **Building a Data Dictionary**
- **Using the Relational Algebra**
- **Using a Structured Query Language**



Using a relational DBMS on the in-class PC.

INTEGRATED COMPUTER SYSTEMS

ENROLL NOW!

Call Toll Free **(800) 421-8166** or

East Coast Office **(703) 847-6448**

West Coast Office **(213) 417-8888**

UB

8805 UB 175

Real-Time System Design: A Hands-On Workshop

HANDS-ON

Course 175



4 Days

BOSTON
June 7-10

SAN DIEGO
June 21-24

WASHINGTON, D.C.
June 28-July 1

LOS ANGELES
July 26-29

BOSTON
August 23-26

TORONTO
September 20-23

WASHINGTON, D.C.
September 27-30

LOS ANGELES
November 1-4

Courses at Your Facility
Please Inquire

Course Benefits

Today's designers are called upon to produce a wide range of commercial and military systems that depend on real-time responses. To successfully complete such designs requires a practical knowledge of both hardware and real-time operating system and applications software, including fixed/adaptive scheduling algorithms and pre-emptive multi-tasking systems.

This unique HANDS-ON course guides the real-time designer through the techniques necessary to analyze and implement real-time systems. Case studies in fields such as telecommunications and process control are used throughout the course to illustrate typical decisions which must be made, and the critical tradeoffs and options which are available.

YOU WILL LEARN HOW TO:

- **Analyze the Time-Critical Aspects of an Application**
- **Segment an Application for Multi-Tasking Operation**
- **Define, Prioritize & Coordinate Application Tasks**
- **Configure Real-Time Kernels & Operating Systems**
- **Write Device Drivers & Interrupt Handlers in the C Language**
- **Integrate, Test and Debug a Real-Time System**

WHO SHOULD ATTEND:

This course is valuable for software and hardware engineers, programmers, systems analysts and technical managers who are involved in specifying, designing or implementing real-time, microprocessor-based systems. A background in programming with high-order languages is assumed.

Course Content

1. OVERVIEW OF REAL-TIME SYSTEMS

Fundamental Concepts
- What Does "Real-Time" Mean?
- Implications of Concurrent Processing
- How Multi-Tasking Systems Work
- Capabilities of Kernels & Real-Time Operating Systems (RTOSs)

Steps in Implementing a Real-Time System
- Analyzing Time-Dependencies
- Configuring and Interfacing the Hardware
- Selecting a Real-Time Operating System
- Developing the Multi-Task Solution
- Integrating and Testing the System

2. HANDS-ON INTRODUCTION TO RTOSs

Using The In-Class Development System
- Directory Functions and File Structure
- Running the C Compiler
- Using the Debugger and Utilities
- Interfacing to the BIOS and DOS
- Running the RTOS Application Routines

Writing Simple Real-Time Routines in the C Language
- Creating Real-Time Tasks
- Inter-Task Communication Techniques
- C Language Input/Output Routines

3. ANALYZING REAL-TIME REQUIREMENTS

Identifying Potential Time-Conflicts
- Contention for CPU, Bus, I/O, & Memory
- Resolving Race Conditions
- The Mutual Exclusion Problem
- Deadlock & Starvation/Lockout Syndromes

Application Timing Analysis Methods
- Using CASE Tools: Advantages/Limitations
- Time Line Charts ■ Critical Region Analysis
- Calculating Available T-States
- Burst Interrupt Problems ■ Stacking
- Full vs. Partial Context Switching

4. USING RTOS SOFTWARE

Defining and Controlling the Tasks
- Assigning Activities to Tasks
- Determining the Task Priorities
- Creating Task Control Blocks
- Specifying the Task Scheduling Method
- Creating, Suspending & Resuming Tasks

Implementing Inter-Task Communication
- Synchronizing the Task
- Using Mail Boxes for Inter-Task Messages
- Flags ■ Semaphores ■ Rendezvous

Controlling Memory Resources
- Allocating and Deallocating Memory
- Fragmentation and Garbage Collection
- Controlling Task Access
- Using Memory Management Chips

Extension of RTOS Facilities
- "Hooks" for High-Order Languages
- Application-Specific System Extensions
- Writing Custom Interrupt Drivers
- Handling Errors and Exceptions

5. RTOS TRADEOFFS AND SELECTION

The RTOS Build vs. Buy Decision
- Simple Applications: Writing a Small, Fast Custom Kernel to Optimize Speed & Memory
- Medium-Scale Applications: Using Off-the-Shelf RTOS Kernels
- Complex Applications: Installing a Full-Scale Real-Time Operating System
- Tradeoffs: Execution Speed, Support Features, Development Time, Volume, Cost

Evaluation of Off-the-Shelf RTOSs
- Functionality ■ Speed ■ Configurabilty
- Modularity/Extendability ■ HOLs
- Facilities for Real-Time Debugging
- RMX-86/286/386/RMK-386
- RMS 68K ■ PSOS
- VRTX ■ MTOS ■ QNIX ■ Others

Languages for Real-Time Applications
- C ■ Modula 2 ■ Forth
- FORTRAN ■ Ada

6. REAL-TIME SYSTEM ARCHITECTURE

Interrupt and Timing Implementations
- Internal Timers for Synchronous Control
- HW/SW Techniques for Interrupt Handling
- Vectored and Priority Interrupt Structures

Handling High Data Rates
- Headroom (CPU Loading) ■ Offloading
- Table Look-Up vs. Algorithmic Processing
- The Partial (Deferred) Solution Method

7. INTEGRATING THE APPLICATION SYSTEM

Real-Time System Integration and Test
- Simulation of Real-Time Software
- Testing Before Integration Begins
- Testing Interfaces, Buses and Controllers
- Verifying Time-Critical Performance
- Isolating Rare-Occurance Real-Time Bugs

Using Diagnostic and Development Tools
- Applying Timing and State Logic Analyzers
- Constraints of In-Circuit Emulators
- Using Software Development Systems

Performance Analysis Tools
- Sampling Analysis ■ Trace Storage ■ IPAT

HANDS-ON

Throughout participants are provided with IBM PC-XTs for developing a series of in-class RTOS applications exercises.

These exercises provide participants with actual experience in performing each of the key steps in designing a real-time system, including:

- **Configuring a Real-Time Multi-Task Operating System**
- **Creating and Prioritizing Tasks**
- **Implementing Inter-Task Communication with Semaphores & Mailboxes**
- **Optimizing Task Scheduling**
- **Dynamically Allocating Memory**
- **Using a Real-Time Debugger to Analyze System Performance**



Participants Developing RTOS Applications on In-Class PC-XTs.

INTEGRATED COMPUTER SYSTEMS

ENROLL NOW!

Call Toll Free **(800) 421-8166** or

East Coast Office **(703) 847-6448**

West Coast Office **(213) 417-8888**

COURSE AUTHORS AND INSTRUCTORS*

C Programming p. 2

Advanced C Programming p. 3

UNIX Workshop p. 5

Mr. Greg Davidson has 8 years of experience with UNIX and C implementations on systems ranging from microcomputers to mainframes, and has had the responsibility for porting both UNIX and C to new hardware configurations. He participated in the development of both PASCAL and Ada, and is the author of two popular program language texts.

Other Faculty: Dr. W. Appelbe, Mr. A. Atlas, Mr. J. Brockmeyer, Mr. W. Buford, Mr. I. Darwin, Mr. M. Fryd, Mr. M. Hammond, Mr. D. Klein, Mr. C. Matthews, Mr. J. McDermott, Mr. J. Ostlund, Mr. R. Rother, Mr. C. Wagar and Mr. R. Zimmermann.

Ada® Programming p. 4

Dr. Bill Appelbe has an extensive background in developing compilers, system support and applications software for Ada, PASCAL, C, and UNIX. Over the last five years, he has worked on Ada compiler development, and a run-time support package for a validated Ada compiler. He also consults to a wide variety of organizations, including IBM, TI, GA, Telesoft, Linkabit, the U.S. Navy, and Los Alamos Labs.

Other Faculty: Ms. L. Bennett, Mr. R. Causey, Dr. S. Franklin, Mr. M. Hammond, Dr. R. Hofkin and Mr. K. Siyan.

SW Requirements & Specifications p. 6

Dr. Merlin Dorfman has 20 years of experience in software development, including all phases of the software life cycle: concept development, requirements analysis, design, implementation, integration & test, and operations & maintenance. His background includes senior systems engineering positions on large real-time data processing systems, and leadership of technology development efforts such as building computer-based tools to aid in systems engineering.

Other Faculty: Ms. P. Hurst and Mr. M. Wallace.

Software QA & Testing p. 7

Ms. Claire Lohr has 20 years of experience with all aspects of system development, from requirements definition through the end of the maintenance phase of the SW life-cycle. She is president of Lohr System Solutions, a firm specializing in Quality Assurance and independent acceptance testing for both the government and private industry, whose clients have included MITRE, Softech, CONTEL, DoD and the FAA.

Other Faculty: Dr. M. Dorfman, Ms. P. Hurst, Mr. B. Larman, Mr. J. Rakos, Mr. G. Tier and Mr. M. Wallace.

Software Maintenance p. 8

Ms. Jody Steinbacher has over 15 years experience in the development, maintenance, and management of software systems. She supported the Space Shuttle Program for over 8 years, including the development of Backup Flight System software. Ms. Steinbacher's application background includes technical management of software development, maintenance activities, and consulting services, as well as work on improving standards for software and information systems.

Other Faculty: Mr. M. Hill, Ms. P. Hurst, Mr. C. Larson, Mr. M. Qualls and Mr. J. Weigant.

Software/User Interface Design p. 9

Dr. Charles Crowley has over 13 years of experience in the design, development and analysis of software/user interfaces. His application experience includes the implementation of improved UNIX operating system interfaces, and work on user interfaces for the Xerox Star System. His current activities include the applications of advanced programming language tools and the development of multi-window, multi-file, mouse-based text editors/word processors.

Other Faculty: Ms. P. Billingsley, Dr. P. Jamar, Ms. K. Jensen, Mr. D. Keller, Mr. P. Kemmerling, Mr. J. Reiley and Mr. W. Stubblefield.

Structured Design & Programming p. 10

CASE: Computer-Aided SW Engineering p. 11

Mr. Richard Zahniser, president of RIX Software Engineering, has over 20 years of experience in designing and implementing real-time transaction processing, communications and process control systems for aerospace, industrial, and manufacturing applications. He has led the development and application of real-time structured methodologies to computer networks and peripheral device controllers.

Other Faculty: Mr. R. Causey, Mr. J. Thomas, Mr. C. Wagar, Mr. M. Wallace and Mr. J. Weigant.

Computer Graphics p. 12

Dr. Edward Angel has implemented computer graphics systems for over 15 years. His application experience includes real-time graphics simulation, computer-aided design, and the development of raster-scan graphics workstations. He is currently involved in the design of knowledge-based graphics systems for robot vision.

Other Faculty: Dr. M. Bailey, Dr. K. Booth, Mr. J. Burk, Dr. J. Brayer, Dr. M. Henderson, Ms. D. O'Shay and Mr. J. L. Schulmann.

Relational Database Systems p. 13

Ms. Lynda Whitley has over 20 years of experience in the design, development, implementation and maintenance of mainframe, mini and microcomputer software for organizations such as Boeing, Control Data and Cray Research. Her practical background has included work on operating systems, application programs, and the design and implementation of end-user systems based on a variety of databases.

Other Faculty: Mr. E. Anttila, Dr. J. Larson and Mr. M. Stiefel.

Real-Time Systems Design p. 14

Mr. Philip Lieberman has extensive experience in the design and implementation of μP applications using 8086, 68000, Z8000, and LSI-11 processors. He has developed real-time multi-tasking operating systems and applications software, and is an active consultant to industry in the design of real-time systems for the control of time-critical processes.

Other Faculty: Mr. W. Brown, Mr. E. Dillingham, Mr. J. Ramsey, Mr. M. Shinn and Mr. C. Wagar.

*Each course is taught by the author or one of a team of highly skilled instructors.



Learn from Experts

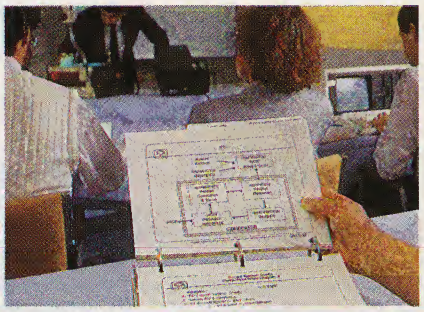
These courses are developed and taught by technical professionals . . . for technical professionals. You'll benefit from a well-organized presentation that emphasizes the practical application of state-of-the-art technology.



Effective Methods

Dual overhead projectors insure rapid communication. Where appropriate, slides, films, videotapes and live demonstrations augment the presentation.

In many courses, computer-based training with hands-on exercises, lab sessions and small team workshops provide concrete "Learn-by-Doing" experience.



Comprehensive Materials

Comprehensive Course Notes replicate each projected transparency, permitting full concentration on what is being taught.

The "State-of-the-Art Supplemental Readings" amplify key course topics, and provide a valuable "after course" reference.

PRODUCTIVITY THROUGH EDUCATION®

UB

13 COURSES ON SOFTWARE DEVELOPMENT

HOW TO ENROLL

PHONE:

TOLL FREE: (800) 421-8166

INTEGRATED COMPUTER SYSTEMS

East Coast Office
(703) 847-6448

West Coast Office
(213) 417-8888

WRITE:

8000 Towers Crescent Dr.
Suite 300
Vienna, VA 22180
Fax: (703) 847-6319

5800 Hannum Ave.
P.O. Box 3614
Culver City, CA 90231-3614
Telex: 4995168 ICS LA
Fax: (213) 410-2952

Phone Today to Reserve Your Place:
Class sizes are limited and classes are often filled to capacity. We recommend that you enroll early.

Immediate Confirmation:
You will receive immediate confirmation of enrollment and course location details by phone (or telex) and subsequently by mail. You can reserve your place without obligation. There is no cancellation fee if you notify us at least two weeks before the course. If a course is cancelled for any reason, you will be notified at least two weeks in advance.

$100 Prepayment Discount:
While our long-standing policy is that payment is not required until after you attend the course, we are pleased to offer a $100 discount for prepayment. Payment may be made by check or major credit card: VISA MasterCard AMERICAN EXPRESS

$100 Multiple Enrollment Discount:
A $100 discount for each participant is provided when 2 or more from the same organization enroll and are invoiced together. Each gets the discount even if he or she enrolls for different courses or different sessions of the same course. PLEASE NOTE: This discount is available *in addition* to the $100 prepayment discount.

Please take advantage of these discounts and call today to enroll.

Course Fee:
The course fee includes a comprehensive set of course materials, and morning and afternoon refreshment breaks each day. These course materials include a detailed five-level set of course notes that parallels the presentation, case studies, all workshop session exercises, and a set of State-of-the-Art Reprints.

Course	Title	Length	Discount Prepaid Fee	Standard Fee
175:	Real-Time System Design: A Hands-On Workshop HANDS-ON	4 Days	$1295	$1395
306:	Software/User Interface Design DEMO LABS	4 Days	$1195	$1295
311:	Software Requirements & Specifications	3 Days	$995	$1095
312:	Software Quality Assurance & Testing	3 Days	$995	$1095
314:	Software Maintenance	3 Days	$995	$1095
320:	Structured Design & Programming	4 Days	$1195	$1295
324:	CASE: Computer-Aided Software Engineering HANDS-ON	4 Days	$1195	$1295
335:	Ada® Programming & Software Engineering HANDS-ON	4 Days	$1295	$1395
336:	UNIX Hands-On Workshop HANDS-ON	4 Days	$1295	$1395
338:	C Programming: Hands-On Workshop HANDS-ON	4 Days	$1295	$1395
339:	Advanced C Programming: Hands-On Workshop HANDS-ON	4 Days	$1195	$1295
365:	Computer Graphics Software & Hardware DEMO LABS	4 Days	$1195	$1295
382:	Relational Databases: Design, Tools & Techniques DEMO LABS	4 Days	$1195	$1295

Course Hours & Completion Certificate:
Orientation (first day): 8:30am–9:00am
Class Hours: 9:00am–4:30pm (last day 9:00am–3:30pm)
Informal Discussion with Instructor: 4:30pm–5:30pm

Each attendee receives a diploma awarding a Continuing Education Unit (CEU) for each 10 hours of class participation.

BRING THESE COURSES TO YOUR FACILITY



▲ ***WHEN PRESENTED AT YOUR OWN FACILITY, THESE COURSES*** *provide even greater cost savings. Tuition is reduced, and travel expenses are eliminated, making it cost-effective to train entire teams.*

You'll Benefit From . . .
Proven, successful courses with expert instructors and comprehensive materials. For Hands-On and Demonstration Courses, we bring all of the in-class computers and electronic equipment to your facility.

A Major Advantage
Courses at your facility can be customized. Specific topics can be emphasized according to the needs of your organization, and timing can be adjusted to fit your schedule. By training an entire team, you'll accelerate the beneficial impact of new technology within your organization.

Experience Produces Results
Over the last 12 years, we have presented thousands of On-Site Courses for hundreds of organizations including: IBM, AT&T, GE, the US Navy, Tektronix, Hughes, HP, and TI. These and other references are available.

Take the First Step Today!
Courses at your facility are surprisingly easy and quick to organize. The first step can be taken by the technical staff, training department, or management.

For Courses at Your Facility Phone:
Anne-Lyse Heron or Reta Singer
On-Site Coordinators
(800) 421-8166 or
(213) 417-8888

We'll provide you with all the assistance and information you need to bring these high-impact courses to your organization.

Call today for a complete set of three brochures with details on 39 Technical Short Courses: (800) 421-8166.

Priority Code:
To facilitate enrollment, please give our registrar the Priority Code:

PCODE 5735



To Order Course Materials Separately:
Each participant receives a comprehensive set of course materials included in the course fee. If you are unable to attend, you may purchase the course materials separately.

Price: $275 per course (includes regular UPS shipping within USA)

Sales Tax: California residents please add 6% (6½% in Los Angeles County)

© Copyright Integrated Computer Systems 1988.
8805UB

Integrated Computer Systems Presents . . .



100 SCHOLARSHIPS



We are proud to announce the opening of our WASHINGTON, D.C. EDUCATION CENTER

To celebrate we are offering you the chance to win any of 100 scholarship opportunities, including all expense-paid attendance at the Integrated Computer Systems course of your choice offered at our new Washington, D.C. Education Center!

5 All Expense-Paid Scholarships

Participate in the course of your choice at our new Washington, D.C. Education Center! You'll receive TUITION-FREE ATTENDANCE at the Integrated Computer Systems course of your choice, ROUND-TRIP AIRFARE to Washington, ACCOMMODATIONS AND PER DIEM for six days.

10 Full Tuition-Paid Scholarships

Attend the course of your choice at any Integrated Computer Systems course site location!
You'll receive TUITION-FREE ATTENDANCE to the course of your choice at any of our course locations in the U.S. or Canada.

85 $100 Tuition Credit Scholarships

You'll receive a $100 credit towards the tuition for attendance to the course of your choice at any of our course locations in the U.S. or Canada.

THREE EASY WAYS TO ENTER:

ENTRY FORM: Complete the Scholarship Entry Form below, detach and mail it to us today. No course enrollment or other purchase is required to participate. OR . . .

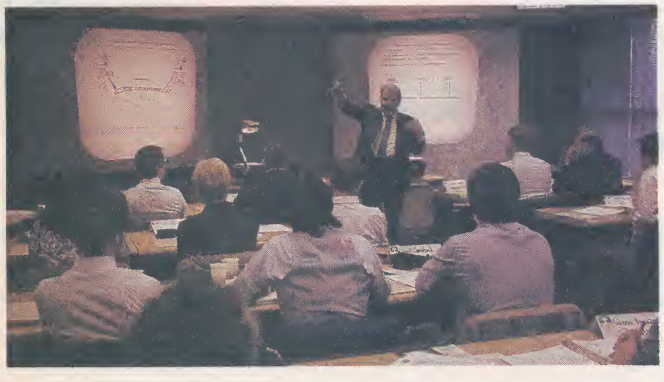
ENROLL: Enroll before July 1, 1988 and we'll automatically enter you in the scholarship drawing. So enroll today! OR . . .

PHONE: You can enter the scholarship drawing by calling us toll-free at **800-421-8166**. Our Customer Service Representatives will be happy to fill out an entry form for you.

YOUR ENTRY MUST BE RECEIVED BY JULY 1, 1988 SO CALL OR WRITE US TODAY.

SCHOLARSHIP ENTRY FORM

- ☐ **YES!** I want to enter the 100 SCHOLARSHIPS Drawing.
- ☐ **YES!** I want to continue to receive brochures on . . .
 - ☐ Software Development Courses
 - ☐ Datacomm, Networks & Artificial Intelligence Courses
 - ☐ Engineering Courses
 - ☐ Management Courses
- ☐ **YES!** Send me information about having these courses presented at my facility.



ENTER THE SCHOLARSHIP DRAWING by either enrolling in the course of your choice, phoning us at 800-421-8166, or returning this Entry Form.

AFFIX LABEL HERE

Please affix address label here. This is absolutely VITAL since it permits us to efficiently locate your name. Note: If you received multiple envelopes, return all labels indicating which is to be kept and which is to be deleted.

TO CHANGE YOUR ADDRESS:

If address on label is incorrect, please affix label and write in corrected address below.

The course of my choice is:

_____ _____
Course Name Course No.

Please complete the following skill testing exercise as required by regulation in certain areas.

(_____ × _____) + _____ = _____
Course No. you selected above. No. of days your selected course lasts. Page No. your selected course appears on.

The priority code appearing on the back page of the brochure is: _____

My phone number is: () _____

Mr. Ms.
Dr. Mrs.
Miss

First Name Last Name

Title

Company Mail Stop

Address

City State Zip Code

UB8805

OFFICIAL SCHOLARSHIP DRAWING RULES

1. To participate in the "100 Scholarships" drawing: (a) enroll in an Integrated Computer Systems course between March 1, 1988 and July 1, 1988, and provide all information requested on the entry form or (b) submit an entry form, complete with your name, title, company name, address, phone number, priority code, skill test answer, and state the course of your choice, or (c) submit a reproduction of the entry form, or (d) submit a 3 x 5 card stating the above information, or (e) give us this information by telephone. All entries must be complete. Only one entry per person is permitted.
2. Entries must be received no later than July 1, 1988. Integrated Computer Systems is not responsible for lost, late, damaged, or misdirected mail.
3. Scholarship drawing is open to residents of the United States and Canada, except where prohibited by law and regulation. Odds of winning depend upon the total number of eligible entries received.
4. By entering the scholarship drawing, all participants agree that all scholarships are awarded upon the condition that Integrated Computer Systems and the employees therof, will have no liability whatsoever for any injuries, losses, or damages of any kind resulting from the acceptance, possession, or use of the scholarship offer.
5. Scholarship recipients will be selected from all eligible entries received in a blind drawing conducted by Advanced Technology Marketing on July 15, 1988. Recipients will be notified by mail and may be required to sign an Affidavit of Eligibility and Release, which must be returned within 14 days of notification or eligibility will be forfeited.
6. If Integrated Computer Systems is unable to contact the selected scholarship recipient, an alternate recipient will be selected by random drawing. No scholarship substitutions will be allowed except as and where necessary due to availability. No recipient will be awarded multiple scholarships. All 100 scholarships offered will be awarded. Taxes on all scholarship offers are solely the responsibility of the recipient.
7. No course enrollment or other purchase is required to participate.
8. Scholarships Offered:

 (5) All Expense-Paid Scholarships — Recipients receive one tuition-paid attendance at one of the scheduled courses at the Washington, D.C. Education Center. Includes: Round-trip coach airfare for one (arrangements will be made through Integrated Computer Systems' travel agency) from any major U.S. or Canadian commercial airport closest to recipient's home (may involve connecting flights); round-trip ground transportation from Washington, D.C. airport to the Tyson Corners Marriot; single occupancy hotel accommodations at the Tyson Corners Marriot for six nights; and maximum daily expenses of up to $40 for six days. Course attendance and all scholarship considerations must be completed by December 31, 1988. Approximate scholarship value: $2600.

 (10) Full Tuition-Paid Scholarships — Recipients receive one tuition-paid attendance at the Integrated Computer Systems course of choice at any of the U.S. or Canadian scheduled locations. Approximate scholarship value: $1395.

 (85) $100 Tuition-Credit Scholarships — Recipients receive a credit of $100 towards attendance at the Integrated Computer Systems course of choice.

NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES

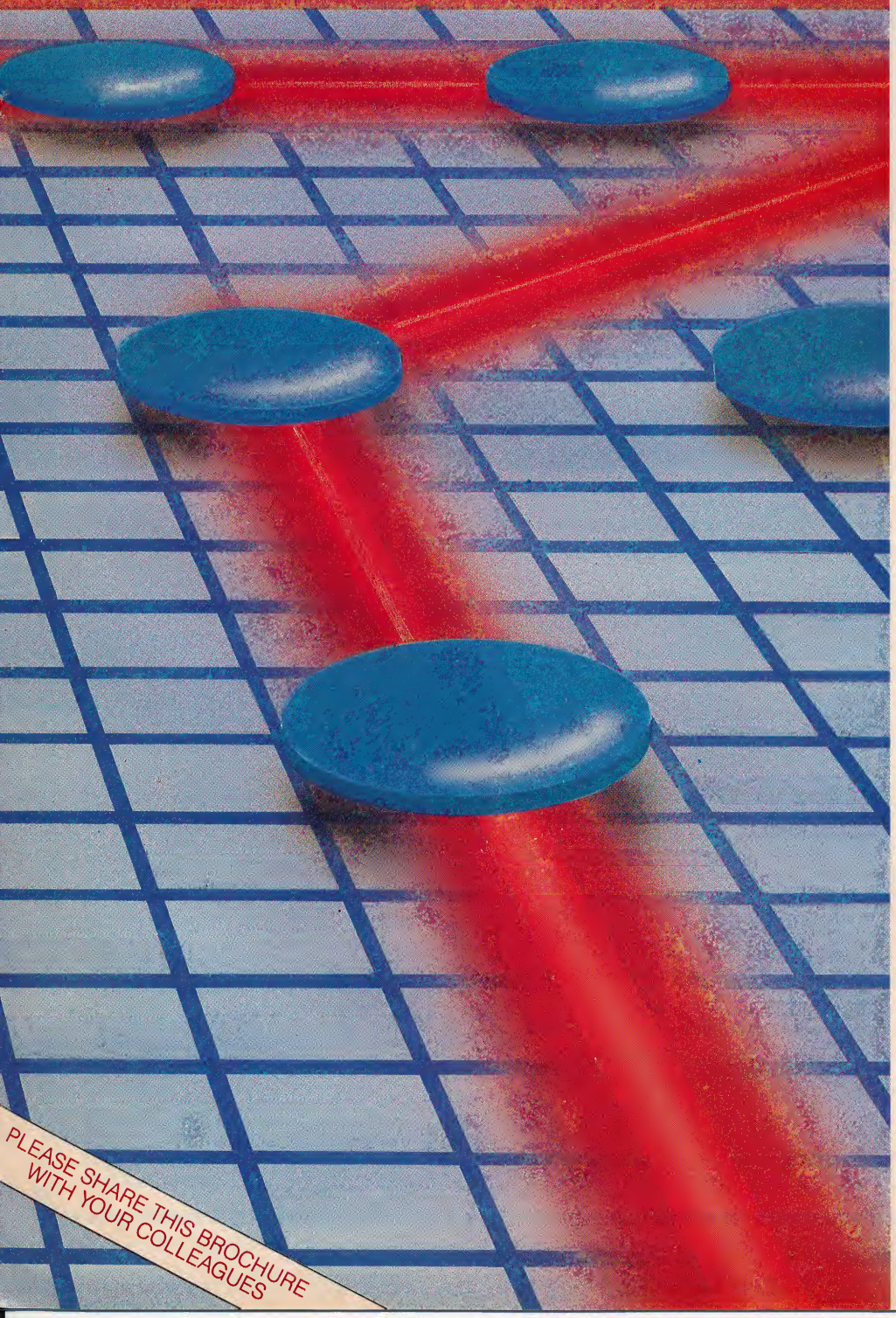
BUSINESS REPLY MAIL

FIRST CLASS MAIL PERMIT NO. 756 CULVER CITY, CA

POSTAGE WILL BE PAID BY ADDRESSEE:

INTEGRATED COMPUTER SYSTEMS
5800 Hannum Avenue
P.O. Box 3614
Culver City, CA 90231-9945

DATACOMM & NETWORKS PLUS ARTIFICIAL INTELLIGENCE COURSES



PLEASE SHARE THIS BROCHURE WITH YOUR COLLEAGUES

SHORT COURSES

May 1988-December 1988

Artificial Intelligence

- Knowledge-Based Systems & Artificial Intelligence HANDS-ON 2
- Expert Systems Design & Development: Hands-On Workshop HANDS-ON 3

Computer Networks

- Introduction to Datacomm & Networks 4
- Troubleshooting Datacomm & Networks: Hands-On Workshop HANDS-ON 5

NEW! Computer Network Management & Control 6

- Computer Network Design: Protocols & Layered Architectures 7
- Micro–Mainframe Connectivity: Practical Linking Tools & Techniques DEMO LABS 8

LANs

- Local Area Networks: Implementation & Configuration 9
- PC Networking: Hands-On Workshop HANDS-ON 10
- MAP/TOP: Technical Introduction & Implementation Status Report . . . 11

Communications

- Fiber Optic Communication Systems: Design Tools & Techniques 12
- Integrated Voice/Data Communications & ISDN 13
- Digital Communication Systems: Design & Applications 14

Enrollment & On-Site Courses 16

from the learning tree®

INTEGRATED COMPUTER SYSTEMS

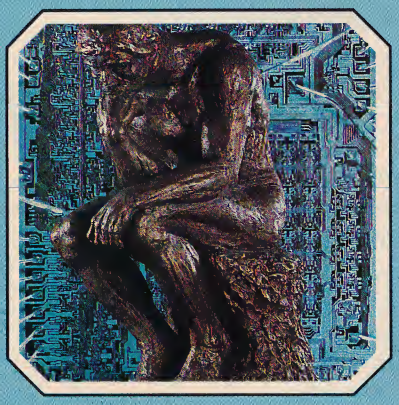
EDUCATION IS OUR BUSINESS®

8805UB

Knowledge-Based Systems & Artificial Intelligence

HANDS-ON

Course 390



4 Days

PALO ALTO
June 7-10

BOSTON
June 14-17

WASHINGTON, D.C.
June 21-24

SAN DIEGO
July 12-15

WASHINGTON, D.C.
August 9-12

SAN FRANCISCO
September 13-16

WASHINGTON, D.C.
September 27-30

SAN DIEGO
October 25-28

WASHINGTON, D.C.
November 29-December 2

Courses at Your Facility
Please Inquire

8805 UB 390

Course Benefits

Knowledge-based systems have emerged from government, industry and university labs and are being successfully applied in a growing number of areas. Powerful tools, ranging from special-purpose hardware to general-purpose languages and software, are commercially available to assist in the development of knowledge-based systems.

This course provides a comprehensive introduction to the key practical principles, techniques and tools being used to implement knowledge-based systems. Each participant gains **extensive hands-on experience** in designing AI representations and search strategies through **in-class computer-based exercises** that provide concrete illustrations of the concepts and techniques presented.

YOU WILL LEARN HOW TO:

- **Understand the Key Principles of Artificial Intelligence (AI) and Their Use in Knowledge-Based Systems**
- **Explore the Efficiency of Different Knowledge-Based Inference Rules and Search Strategies for Specific Applications**
- **Evaluate Available Languages, Tools, Assistance Packages and "Skeleton" Systems to Aid Application Implementation**
- **Implement AI Concepts During Hands-On Workshops**

WHO SHOULD ATTEND:

This course is valuable for all technical professionals and their managers who require a comprehensive overview of the state-of-the-art in knowledge-based systems. A technical orientation and some background in programming will be helpful.

Course Content

1. INTRODUCTION

Basic Concepts and Terminology
- What Is Artificial Intelligence?
- The Relationship of AI, Intelligent Machines and Expert Systems
- Symbolic Representation of Knowledge
- Heuristic Search and Inference: How a Knowledge System "Thinks"

Applications for Knowledge-Based Systems
- CAE ■ Office Automation
- Signal and Image Interpretation
- Monitoring/Controlling Complex Processes
- Natural Resource Analysis ■ Robotics

Overview of Representative R&D Activities
- Governmental/Institutional: DARPA (SCS), MCC, SRC, MCNC, Rand, SRI, ICOT, Others
- Universities: MIT, Stanford, CMU, Others
- Commercial: Xerox, Schlumberger, DEC, IBM, HP, GE, RCA, TI, Bell, Others

2. FUNDAMENTALS OF KNOWLEDGE ENGINEERING

The Role of Predicate Calculus
- Understanding Predicate Calculus
- Basic Components: Terms, Predicates, Functions, Connectives, Quantifiers
- Machine Reasoning with Rules of Inference
- Resolution and Other Rules of Inference

Knowledge Representation
- Trees and Graphs ■ Static Decision Trees
- Network Representations
- Rules and the Dynamic Tree
- Frame and Object-Oriented Simulation

Simple Systems Using Production Rules
- Representing Knowledge with Procedures
- IF-THEN Production Rules
- Productions that Generate Trees
- Pattern Matching/Unification Techniques

Searching for Solutions
- Forward Chaining, Data-Driven Reasoning
- Backward Chaining, Goal-Driven Reasoning
- Depth First vs. Breadth First Search
- Graph Search Algorithms

Heuristic Search Methods and Strategies
- Using Domain-Specific Information
- Mini-Max and Alpha-Beta Pruning
- Best First Search
- Hierarchical Generate and Test
- Intelligent Backtracking

Reasoning with Uncertainty
- Abductions: An Alternative to Deduction
- Certainty Factor Algebras
- Bayes, Zadeh and Other Approaches
- Default Reasoning
- The "Closed World" Assumption

Knowledge Structuring Techniques
- Scripts ■ Schemas
- Blackboards ■ Frames
- Conceptual Dependency Networks
- Semantic Network Representation

3. TOOLS FOR SYSTEM DEVELOPMENT

LISP: Linked List Processing
- Overview of Common LISP and Other Dialects
- Basic LISP Functions
- Writing a Simple LISP Program

PROLOG: Programming in Logic
- Overview of Features ■ Resolution
- Separation of Logic and Control
- Writing a Simple PROLOG Program

Smalltalk: Object-Oriented Programming
- Relationship to LISP and PROLOG
- Using Objects to Organize Problem Solving
- Inheritance ■ Windowing ■ Hierarchies

Tools for Designing Expert Systems
- Prototyping: Exploratory Programming
- Shells and Skeletons: EMYCIN, KAS, EXPERT
- Languages: OPS5, ROSIE, HEARSAY-III
- Other Tools: TEIRESIAS, AGE, IPE, LOOPS

Computers for Expert Systems
- General- vs. Special-Purpose Computers
- 3600 (Symbolics)
- Dolphin (XEROX) ■ Daisy
- Explorer (TI) ■ Tektronix 4404 ■ Others

4. BUILDING KNOWLEDGE SYSTEMS

Knowledge-Based Control for Robots
- Defining the Robot's World
- Use of Pre- and Post-Condition Sets
- System Integration and Results

The VAX Configuration Expert: XCON
- Acquisition of Expert Human Knowledge
- Implementation Using OPS
- Machine Performance and Cost Benefits

Potential Military Applications
- Autonomous Vehicles ■ Pilot's Associate
- Battlefield Management Models

Mycin: The Medical Expert
- Reasoning with Uncertainty
- TEIRESIAS: Knowledge-Base Editor

5. PRESENTATIONS & WORKSHOPS

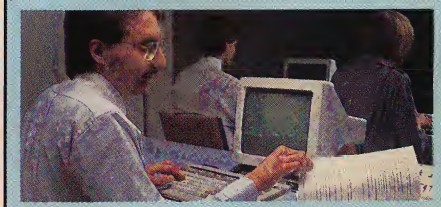
- KEE (FIXER) ■ ART (NAVEX) ■ STEAMER
- TEKTRONIX (Smalltalk) ■ Others
- Search Algorithms ■ Production Systems
- Sentence Parsing ■ Expert System Shell
- Sample Expert System

HANDS-ON

This course emphasizes **in-class, hands-on exercises** implementing Artificial Intelligence representation and search techniques. Participants have dedicated access to a UNIX-based system throughout the course.

Exercises include:

- **Programming in LISP and PROLOG**
- **Building a Small Expert System**
- **Analyzing the Effects of Alternative Search Strategies**
- **Improving System Performance**



Throughout the course, participants execute in-class exercises using a workstation in a multi-user environment.

INTEGRATED COMPUTER SYSTEMS

ENROLL NOW!

Call Toll Free **(800) 421-8166** or

East Coast Office **(703) 847-6448**

West Coast Office **(213) 417-8888**

8805 UB 392

Expert Systems Design & Development: Hands-On Workshop

HANDS-ON

Course 392



4 Days

PALO ALTO
May 24-27

TORONTO
June 7-10

LOS ANGELES
July 12-15

WASHINGTON, D.C.
July 26-29

OTTAWA
August 23-26

BOSTON
August 30-September 2

WASHINGTON, D.C.
September 20-23

SAN DIEGO
October 25-28

WASHINGTON, D.C.
November 29-December 2

Courses at Your Facility
Please Inquire

Course Benefits

The emergence of Expert Systems software technology has placed powerful **Tools for Knowledge-Based Applications** at the disposal of a wide spectrum of users. These tools permit the solution of practical, real-world problems with development times ranging from days to weeks, often without requiring the use of a programming language.

This course provides a solid foundation in the application of Expert Systems through **EXTENSIVE IN-CLASS HANDS-ON EXPERIENCE under the guidance of a skilled instructor**. Participants will return to their jobs with the experience of having built an Expert System and the ability to evaluate and select Expert System Tools suitable for their applications.

YOU WILL LEARN HOW TO:

- **Use In-Class Computers to Build and Enhance a Knowledge Base**
- **Analyze the Suitability of Applications for Expert System Solutions**
- **Evaluate the Strengths and Limitations of Available Software**
- **Select Appropriate Expert Tools and Shells for Your Applications**
- **Successfully Build Expert System Solutions in a Wide Range of Environments**

WHO SHOULD ATTEND:

This course is valuable for both technical and non-technical professionals and managers who are, or will be, involved in evaluating, selecting or implementing Expert System Applications.

Course Content

1. INTRODUCTION AND OVERVIEW

Basic Concepts and Terminology
- What is an Expert System? A Shell?
- Defining a "Knowledge Base"
- Facts, Rules and Inference Engines
- Identifying Suitable Applications

Features and Benefits of Expert Systems
- Reduced Costs, Enhanced Productivity
- Preserving and Disseminating Expertise
- Improved Timeliness and Quality of Decisions
- Rapid Prototyping and Development

Overview of Applications
- Intelligent Job Aids ■ Automated Advisors
- Diagnostic and Troubleshooting Guides
- Marketing, Sales, Financial and Legal Aids
- Manufacturing and Aerospace Applications

2. HANDS-ON INTRODUCTION

Understanding The Technology
- Effective Use of Expert System Tools
- Acting as Your Own "Knowledge Engineer"
- Getting a Feel for Performance
- Designing an Initial Application

Using the In-Class Computers
- Running an Expert System Shell
- Interacting with the User Interface
- Adding Facts and Rules to the Knowledge Base

3. SOFTWARE ARCHITECTURES FOR KNOWLEDGE ENGINEERING

Expert Systems Based on Decision Trees
- Using Static Tree Methodology
- Building Decision Trees
- Evaluating Tree Decision Points

Rule-Based Expert Systems
- The Stanford Model ■ Production Systems
- Strategies for Controlling Search
- Using Dynamic Trees and Graphs
- Program-Generated Explanation: "Why, How"

Designing With AI Languages
- Comparing Features of: LISP, PROLOG, OPS
- Building the Inference Engine and Rule Base
- Debugging Aids and the User Interface

Frame/Object Based Expert Systems
- Objects as *Active, Parallel and Asynchronous*
- Inferencing Using Inheritance and Messages
- Principal Languages: SMALLTALK, FLAVORS
- Available SW Tools: KEE, ART, PICON, others

4. SEARCH STRATEGIES

Basic Search Techniques
- Searching Trees and Graphs
- Goal Directed vs. Data Driven
- Depth First vs. Breadth First
- Heuristic or Best First Search

Reasoning With Uncertainty
- Deductive vs. Abductive Reasoning
- Utilizing Bayes Rule
- Using the Stanford Certainty Algebra

Building a Solution
- Analysis of Problem Type
- Balancing Search Techniques
- Engineering Compromises and Tradeoffs

5. BUILDING EXPERT APPLICATIONS

System Development Guidelines
- Establishing an Overall Project Plan
- Characterizing the Problem and "Domain"
- Defining, Scoping and Focusing the Solution
- Extracting Information from the "Expert"

Hands-On Case Studies
- Designing the Initial Rule Set
- Developing a Prototype Solution
- Adding, Structuring and Linking Rules
- Setting Confidence Levels and Probabilities
- Iterative Improvement of the Expert System

6. SELECTING AN APPROPRIATE SW PACKAGE

Key Evaluation Criteria
- Overall Power ■ Cost ■ Speed
- Ease of Learning, Development and Use
- Hardware Required: RAM, Disk, Graphics
- Selection Pitfalls to Avoid

Typical Available Shells:
- Expert Edge ■ M.1 ■ Rule Master ■ EXSYS
- ExpertEase ■ Insight 2+ ■ Nexpert
- ES/P Advisor ■ KES II ■ Guru ■ 1st Class
- Personal Consultant Plus ■ Expert Choice

Advanced Topics and Trends

HANDS-ON

This course is UNIQUE in providing participants with extensive guided Hands-On Experience utilizing In-Class Computers and selected Expert System Shells, Tools and AI Languages.

Hands-on exercises include:

- **Developing an Expert System Application**
- **Building and Modifying the Knowledge Base**
- **Interacting with the Expert System**
- **Adding Rules and Facts to Enhance Performance**
- **Testing and Debugging the Application**



Course participants using In-Class Computers and Expert System SW to develop applications.

INTEGRATED COMPUTER SYSTEMS

ENROLL NOW!

Call Toll Free **(800) 421-8166** or

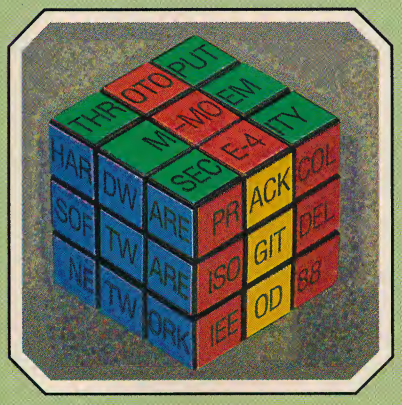
East Coast Office **(703) 847-6448**

West Coast Office **(213) 417-8888**

Introduction to Datacomm & Networks

A Comprehensive Foundation in Technology, Tools & Techniques

Course 350



4 Days

WASHINGTON, D.C.
June 14-17

LOS ANGELES
July 12-15

WASHINGTON, D.C.
July 19-22

SAN DIEGO
August 2-5

BOSTON
August 9-12

SAN FRANCISCO
August 16-19

WASHINGTON, D.C.
August 23-26

WASHINGTON, D.C.
September 20-23

LOS ANGELES
October 4-7

Courses at Your Facility
Please Inquire

8805 UB 350

Course Benefits

The field of data communications is undergoing tremendous change, spurred by the growing availability of sophisticated "off-the-shelf" hardware, software and network components. The challenge of integrating sophisticated data communication technology into products, systems & services means that technical professionals must develop a solid foundation and "state-of-the-art" expertise in data communciations technology.

This course provides an engineering-oriented introduction to data communications and computer network technology. Participants will gain a thorough understanding of how the basic components work, and how they're put together to implement a system.

YOU WILL LEARN:

- **Fundamental Data Communication & Network Concepts**
- **How Data Communications Hardware and Software Components Work from an Engineering Point of View**
- **Key Protocols & Standards**
- **How Packet Switching Networks Operate**
- **How Local Area Baseband & Broadband Networks Function**
- **Strategies for Linking PCs with Minis & Mainframes**

WHO SHOULD ATTEND:

This course is valuable for hardware and software engineers, systems analysts and other technical data processing and telecommunications professionals and managers who require a technical introduction to data communications.

Course Content

1. INTRODUCTION & OVERVIEW

Understanding the Terminology
- Bit Rate ■ Baud
- Frames, Packets & Messages
- Protocols & Networks
- Data Transmission Techniques
- Terminals, Hosts & Modems
- Local Area & Long Haul Networks
- Integrated Data & Voice Communication

Applications Overview

2. FUNDAMENTALS OF DATA COMMUNICATIONS

Transmission Media
- Twisted Pair ■ Coaxial Cable ■ CATV
- Fiber Optics ■ Microwave ■ Satellites
- Bandwidth/Speed/Cost Tradeoffs

Characteristics of Computer Terminals
- Character Sets ■ Buffering Capabilities
- Block vs. Character Transmission
- "Intelligent" Local Processing Capability

Modems & Modulation
- Basic FSK & PSK Modulation Schemes
- Simplex, Half & Duplex Transmission
- High Speed Intelligent Modems
- Bell & CCITT Modems ■ Diagnostics

Bit Serial Transmission of Data
- Synchronous vs. Asynchronous
- Manchester, NRZI & Other Encoding
- Error Detection & Correction Codes
- In-Band Control ■ Clock Recovery

3. NETWORK FUNDAMENTALS

Basic Network Concepts, Models & Standards
- The ISO Layered Network Model for Open Systems Interconnection (OSI)
- The Role of the X.25 Standard
- SNA, DECnet, Ethernet, Wangnet, Others

Network Topologies
- Rings & Loops ■ Bus & Tree Structures
- Centralized Stars ■ Hierarchical Designs
- Centralized vs. Distributed Control

Effect of Topology on Network Performance
- Throughput ■ Ease of Configurability
- Vulnerability to Malfunction ■ Delay

Packet Switching Techniques
- The Structure of a Packet
- Store-and-Forward Packet Switching
- Direct (Common Media) Packet Switching
- Packet Assembler & Disassembler (PAD)

4. PROTOCOLS & NETWORK DESIGN

The Functions of Data Delivery Protocols
- Addressing ■ Synchronization
- Flow, Sequence & Error Control
- Circuit Management ■ Security
- Virtual Circuit vs. Datagram Approaches

The Physical/Electronic Layer
- The Electronic Interface Between Terminals, Modems, Media & Hosts
- RS-232C Details: Signal & Control Lines
- Standards: V.24, X.21, RS-449
- FDM & TDM Multiplexers ■ Concentrators

The Data Link Layer & Protocols
- Defining the Data Block/Frame Structure
- Byte-Oriented Protocols: Bisync, DDCMP
- Bit-Mode Protocols: HDLC, SDLC, ADCCP
- Flag, Address, Control, & Check Fields
- VLSI Multi-Protocol Chips

The Network Layer
- Virtual Circuit vs. Datagram Services
- Routing Strategies, Advantages & Pitfalls
- Fragmentation & Reassembly of Packets
- Flow & Congestion Control Across the Net
- Handling Errors & Outages

Higher Layers
- Transport ■ Session
- Presentation ■ Application

Interwork Gateways

5. LOCAL AREA NETWORKS (LANs)

Baseband LAN Techniques
- Contention: Carrier Sensed Multiple Access With Collision Detection (CSMA/CD)
- Slotted Rings and Token Passing
- Master Controller Methods

Broadband LAN Techniques
- Dedicated, Switched & Shared Channels
- Mid-Split & Twin Cable Designs
- Frequency Agile Devices

Integrated Voice/Data PABX Systems
- Use of In-Place Cabling
- Hybrid vs. All-Digital Switches

LAN Components
- Cables, Connectors & Taps
- Transceivers ■ Repeaters
- Ethernet & Ring Controller Boards & Chips

Examples of Available LANs
- Baseband: Xerox Ethernet, NET/One, Hyperchannel, Cambridge Ring, Data Point
- Broadband: Wangnet, LocalNet
- Advanced PABX-Systems

6. INTERFACING PERSONAL COMPUTERS TO MINIS & MAINFRAMES

Linking PCs To Remote Computers
- Configuring the Communication Link
- Using Available Packet Switched Networks
- Accessing Information Utilities: SOURCE, Dow Jones, Dialog, Others

Linking PCs to Local Minis & Mainframes
- Emulation Software to Link PCs/Mainframes
- "Interfacing" PC & Mainframe Databases
- Uploading & Downloading Files
- Available LANs for PCs

7. IMPLEMENTATION CONSIDERATIONS

System Planning & Selection Considerations
- Facilities for Network Control & Monitoring
- Traffic Handling Capabilities
- Reliability ■ Ease of Expansion
- Installation & Maintenance

Survey of Products & Systems

Value Added Networks & Services
- Telenet ■ Tymnet ■ Transpac
- Datapac ■ IPSS ■ Euronet

Current Trends & Future Directions
- Impact of US Deregulation & Evolving European PTT Policies
- IBM & AT&T Activities & Future Directions
- ISDN: Integrated Services Digital Network

8. COURSE REVIEW AND SUMMARY

INTEGRATED COMPUTER SYSTEMS

ENROLL NOW!

Call Toll Free **(800) 421-8166** or

East Coast Office **(703) 847-6448**

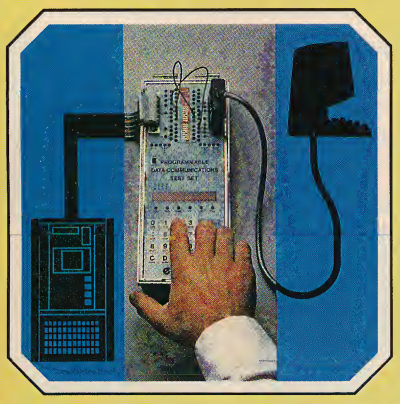
West Coast Office **(213) 417-8888**

Troubleshooting Datacomm & Networks: Hands-On Workshop

8805 UB 356

HANDS-ON

Course 356



4 Days

SAN DIEGO
June 28-July 1
WASHINGTON, D.C.
July 26-29
BOSTON
August 2-5
LOS ANGELES
August 23-26
WASHINGTON, D.C.
September 13-16
PALO ALTO
September 27-30
BOSTON
October 4-7
SAN DIEGO
October 25-28
WASHINGTON, D.C.
November 15-18

Courses at Your Facility
Please Inquire

Course Benefits

Troubleshooting a Data Communication System can be both an efficient and painless process . . . given the proper knowledge, tools and procedures. This course provides fundamental skills and key troubleshooting techniques through **extensive hands-on exercises performed in-class under the guidance of an expert instructor** using typical datacomm test equipment.

Course participants will return to their jobs with the ability to quickly detect, isolate and diagnose problems in their data communications network.

YOU WILL LEARN HOW TO:

- **Use Breakout Boxes, Hand-Held Data Test Sets, Data Analyzers and Other Troubleshooting Equipment**
- **Configure and Troubleshoot Modems and RS-232C Interfaces**
- **Employ Loopback and Bit Error Rate Tests to Isolate Local and Remote Faults**
- **Diagnose Failures in Terminals, Modems, Multiplexers, Switches, Data Links and Networks**
- **Troubleshoot Hardware and Software Problems in the Data Link at the Protocol Level**

WHO SHOULD ATTEND:

This introductory course is valuable for all technical professionals and their managers who are involved in the daily operation, maintenance and troubleshooting of data communication systems and networks.

Course Content

1. OVERVIEW

Basic Concepts and Terminology
- Key Elements of a Datacomm Network
- Half vs. Full Duplex, Synchronous vs. Asynchronous Transmission
- Modems and Modulation Techniques
- Interface and Network Standards

Why Datacomm Systems Fail
- Typical Terminal and Modem Failure Modes
- Device Configuration and Interface Errors
- Data Link Faults: Noise and Degradation
- Multiplexer and Data Switch Malfunctions

Troubleshooting Techniques and Tools
- A Structured Troubleshooting Methodology
- Splitting the Problem into Testable Parts
- Using Test Equipment to Isolate Faults
- Applications of Line and Protocol Analyzers

2. INTRODUCTION TO HANDS-ON TESTING

Modem and RS-232 Fundamentals
- The RS-232 Signal and Control Lines
- Understanding the DTE/DCE Interface
- Key Modem Parameters and Functions
- Fault Isolation and Loopback Testing

Initial In-Class Troubleshooting
- Basic Tools and Test Equipment
- Testing a Terminal–Modem Interface Using a Breakout Box (BOB)
- Diagnosing a Defective Modem Using a Bit Error Rate Tester (BERT)

3. USING TROUBLESHOOTING TOOLS

Employing Hand-Held Testers
- Using BOB and BERT Equipment
- DTE/DCE Emulation and Troubleshooting
- Bit/Character/Block Error Rate Testing
- Testing a Polled, Multidrop System

Conducting Loopback Tests
- Analog vs. Digital Loopbacks
- Local and Remote Loopback Testing
- Conducting End-to-End Tests
- Analyzing Point-to-Point and Multidrop Links

Fault Analysis Techniques
- Detecting DTE/DCE Handshake Problems
- Using "Smart" Cables and Null Modems
- Diagnosing Terminal Failures
- Using Modem Self-Test Features
- Remote Testing Techniques

Troubleshooting Multiplexers and Data Switches
- Modes of Operation ■ Reconfigurability
- Detecting and Diagnosing Common Faults
- "Built-In" Diagnostic Features and Ports
- Flow Control and Load Leveling

4. TESTING TELECOMMUNICATION LINKS

Understanding the Telephone Channel
- Bandwidth ■ Signal Levels
- Line Characteristics and Equalization
- Common Problems and Their Symptoms

Tail Circuits
- Synchronous and Asynchronous
- End-to End Testing
- Monitoring and Trapping

Using Analog Test Equipment
- Key Measurement Techniques
- Line Monitors and Analyzers
- Transmission Impairment Measurements
- Verifying Line Quality

5. TROUBLESHOOTING NETWORK DATALINKS AND PROTOCOLS

Network Architecture Overview
- Dedicated, Switched and Packet Nets
- Protocols in Layered Networks
- Bit and Character Oriented Protocols: Bisync, SDLC, HDLC

Diagnosing Network Software Problems
- Simulating the DTE and the Network
- Decoding Frame/Packet Level Mnemonics
- Packet Delays and Flow Control
- Detecting Errors at the Frame Level
- Setting X.25 PAD Parameters

Using Data and Protocol Analyzers
- Protocol Emulation and Monitoring
- Programming Techniques for Analyzers
- Fault Detection Applications
- Video Demonstrations of BSC and X.25

6. PUTTING IT ALL TOGETHER

Survey of Products and Systems
- Vendor/Device Selection: Navtel, Electrodata, Datatran, Spectron, Metatek, Atlantic Research, Digilog, Halcyon, HP, NEC, ADC, Avant-Garde, Codex, Case
- Pitfalls to Avoid
- Cost/Benefit Analysis

Developing a Network Control Center
- Monitoring and Fault Isolation
- Recovery: Patching vs. Switching
- Specifying Equipment Requirements

Managing the Datacomm Function
- Establishing Performance Standards
- Developing a Troubleshooting Plan
- Maintenance and Spares Philosophy
- Personnel Training ■ Documentation

7. SUMMARY AND FUTURE TRENDS

HANDS-ON

Participants gain extensive Hands-On Experience using datacomm test equipment to isolate and diagnose real world problems.

In-class, hands-on exercises include:

- **Use of a Breakout Box, Bit Error Rate Tester, & Other Test Equipment**
- **Troubleshooting DTE and DCE Interfaces**
- **Fault Diagnosis in Synch/Asynch, Full/Half Duplex Modes**
- **Network Protocol Monitoring and Simulation**



Course participants using test equipment to diagnose datacomm faults.

INTEGRATED COMPUTER SYSTEMS

ENROLL NOW!

Call Toll Free **(800) 421-8166** or

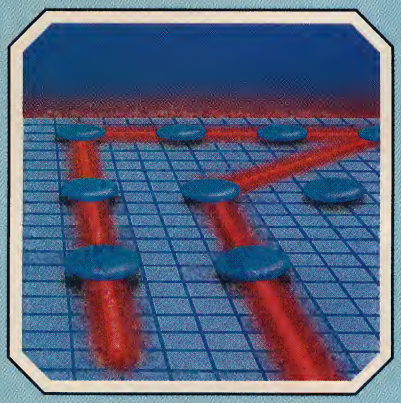
East Coast Office **(703) 847-6448**

West Coast Office **(213) 417-8888**

Computer Network Management & Control

Course 353

NEW!



4 Days

TORONTO
May 31-June 3
SAN DIEGO
June 7-10
WASHINGTON, D.C.
June 21-24
SAN FRANCISCO
July 12-15
BOSTON
July 19-22
WASHINGTON, D.C.
August 9-12
LOS ANGELES
August 16-19
SAN DIEGO
September 27-30
WASHINGTON, D.C.
November 1-4

Courses at Your Facility
Please Inquire

8805 UB 353

Course Benefits

The effective use of network management and control, improves network response time and availability, avoids network failure, and facilitates network growth and development.

This course provides participants with the key tools and skills to manage communication networks in today's environment of rapidly changing network technology. It provides techniques for planning, installation, management, and control of both wide and local area networks. **Emphasis is on the practical aspects of network configuration, start-up and operations, as well as the detection, isolation and correction of network problems.**

YOU WILL LEARN HOW TO:

- **Plan and Manage the Installation/Upgrade of a Network**
- **Monitor and Record the Status of Network Operations**
- **Optimize Network Performance and Availability**
- **Implement Effective User Support Services**
- **Develop Procedures for Detecting and Recovering from Network Failures**
- **Select Network Control and Diagnostic Equipment/Software**

WHO SHOULD ATTEND:

This course is valuable for information systems professionals, telecommunications specialists, network planners and designers, systems analysts, engineers and managers who have an operational or support responsibility for network management and control. Course 350, "Introduction to Datacomm & Networks", or equivalent background, is recommended.

Course Content

1. INTRODUCTION AND OVERVIEW

Key Network Management Functions
- Planning ■ Installation
- Startup ■ Operations and Maintenance

Functions of Network Management
- Service Planning
- Performance Measurement and Optimization
- Network Security ■ User Support
- Failure Detection and Recovery

Tools for Network Management and Control
- Network Operating Systems
- Diagnostic Software and Hardware
- The Network Control Center
- Record Keeping and Error Tracking Tools
- Applications for Expert Systems

Review of Network Technology
- Voice, WAN, and LAN Nets ■ Packet Data
- Modems, Muxes, Nodes, Hosts, Terminals
- Interfaces, Gateways and Bridges
- Protocols and the ISO/OSI Model

2. PERFORMANCE OPTIMIZATION AND SERVICE PLANNING

Determining What To Optimize
- Least Cost Systems
- Minimizing Transit Delay and Response
- Maximizing Throughput
- Controlling Capital and Installation Cost

Optimization Approaches
- Route Optimization Techniques in WANs
- Central vs. Distributed Control
- Spreading the Network Load

Performance Estimation
- Implementation Time Estimation
- Cost, Throughput, Response and Delay
- Measuring Quality and Service Levels

Configuration Methodology
- Parameters Used in Configuration
- Configuring PADs, Clusters and Muxes
- Synchronous Terminal Configuration

Maintenance and Service Planning
- T1, Megastream and CEPT Trunk Services
- Liaison with Carriers and Telcos
- Backup and Network Administration
- Types of Hardware Maintenance Plans

3. INSTALLATION AND TESTING

Site Planning
- Determining Space Requirements/Layout
- Cable and Connection Plan
- Environmental Factors
- Satellite and Microwave Links

Network Implementation Techniques
- Initial Installation vs. Upgrade
- Using the Critical Path Method
- Cutover Plan to Minimize Disruptions

Testing and Startup
- Benchmarks, Trials and Test Planning
- Acceptance Testing
- Network Loading and Stress Testing

4. USER INTERFACING AND SUPPORT

Establishing a User "Help Desk"
- Goals of the User Help Desk
- Monitoring Customer/User Satisfaction
- Fault Validation and Confirmation
- User Education and Training

Inquiry and Support Planning
- Inquiry Check List Construction
- Documentation Principles
- Automation of Inquiry Support
- Inquiry Recording and Statistics

Network Control Personnel
- Recruiting, Training and Motivating
- Structure and Reporting

5. PROVIDING NETWORK SECURITY

Risk Analysis
- Line Tapping and Intercepting
- Hackers, Trojan Horses and Bandits
- Shrinkage ■ Impersonation

Minimizing Security Threats
- Peer Validation and Access Control
- Passwords, Authorization and Audit
- Encryption and Public Key Systems

6. RELIABILITY AND FAULT DIAGNOSIS

Fundamentals of Failure Analysis
- Serial and Parallel Component Failure
- Calculating Availability: MTBF and MTTR
- Bath-Tub and Cyclic Failure Curves
- Software Faults ■ Connectivity
- Elements of Compatibility

Improving Reliability
- Planning Redundancy and Fall-back
- Minimizing Human Error

Fault Isolation
- BERT, BLERT and CCITT V.54 Loop-back
- Using Binary Decision Rules
- History Analysis and Heuristics

Protocol Analysis and Diagnostics
- Interface Timing and Control Signals
- High Level Packet and Protocol Analysis
- BOBS and Hand Held Testers
- Logic and Protocol Analysers

7. THE NETWORK CONTROL CENTER

Network Control Center Functions
- Line and Node Status and Configuration
- Line Diagnostics and Status Monitoring
- Software Control and Distribution
- Fault History Recording

Switching and Display Techniques
- Large and Small System Status Display
- Hand Patching and Switching
- Electro-mechanical Reconfiguration
- Automatic Digital Control

Network Management Tools
- X.25, T1, and Megastream tools
- Features of Network Tools
- Network Managers on LANs and WANs

Management and Control Workshops

During the course participants will gain practical experience through workshop exercises to:

- **Analyze Network Reliability**
- **Make Cost, Performance, and Reliability Trade-offs**
- **Plan Network Fault Testing Procedures**
- **Define User Support Services**
- **Develop a Network Control Center Specification**

INTEGRATED COMPUTER SYSTEMS

ENROLL NOW!

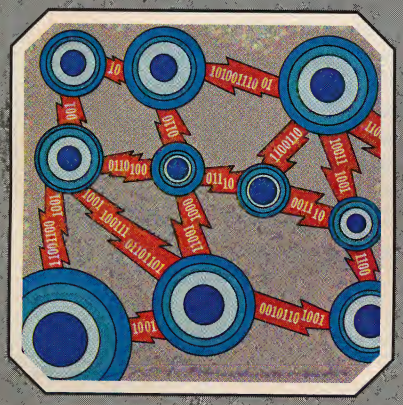
Call Toll Free
(800) 421-8166 or

East Coast Office
(703) 847-6448

West Coast Office
(213) 417-8888

Computer Network Design: Protocols & Layered Architectures

Course 355



4 Days

BOSTON
June 7-10
SAN DIEGO
June 21-24
WASHINGTON, D.C.
June 28-July 1
PALO ALTO
July 19-22
WASHINGTON, D.C.
August 2-5
LOS ANGELES
August 16-19
SAN DIEGO
September 27-30
WASHINGTON, D.C.
September 27-30
SAN FRANCISCO
November 1-4

Courses at Your Facility
Please Inquire

8805 UB 355

Course Benefits

The range and sophistication of Computer Network applications is growing rapidly, propelled by technological advances, standardization, and increasing demand for cost-effective, data communication services.

This course provides an in-depth working knowledge of layered network architectures, protocols, standards, and hardware/software components. **Emphasis is on the practical aspects of packet-switched network design and implementation, utilizing existing and evolving protocol standards.**

YOU WILL LEARN HOW TO:

- **Determine Network Requirements & Perform Design Tradeoffs**
- **Understand the Latest ISO Protocol Developments**
- **Implement Layered Network Communication Protocols**
- **Utilize Packet and Message Switching Techniques**
- **Evaluate Available Network Hardware and Software Components**
- **Use Gateways between Local and Wide-Area Networks**
- **Control a Network to Insure Reliability and Security**

WHO SHOULD ATTEND:

This course is valuable for personnel who are involved in implementing or interfacing with computer communication networks. It will benefit engineers, systems analysts, network designers, communication specialists, programmers and technical managers. Course 350, "Introduction to Datacomm & Networks," or equivalent background is recommended.

Course Content

1. REVIEW OF LAYERED NETWORK ARCHITECTURE

Fundamentals of Network Architecture
- Decomposition of Functions into Layers
- Layer-to-Layer Communication
- Network Links and Nodes

Design of Layered Protocols
- Requirements of Protocol Design
- Data Link and Network Protocols
- Implementing Higher Level Protocols

Network Software and Firmware
- Packet Generation, Routing & Switching
- Addressing ■ Synchronization
- Flow & Error Control

Network Topology
- Physical vs. Virtual Networks
- Hierarchical vs. Mesh Networks

The ISO Reference Model
- Basic Concepts and Terminology
- Layering ■ Services & Functions

2. THE PHYSICAL & DATA LINK LAYERS

Physical Layer Implementation
- Data Transmission Techniques
- Channel Characteristics
- Concentration & Multiplexing

Design of the Data Link Layer
- Maintaining Proper Sequence
- Detecting and Correcting Errors
- Making Control Codes Transparent

Details of the Data Link Protocol: HDLC
- Mode Selection: NRM, ARM, or ABM
- Error Recovery and Exception Handling
- Managing Flow Control: RR & RNR
- Retransmission: REJ and ACK Timeouts
- Effects of Sliding Window Size

Comparison of HDLC with Other Protocols
- SDLC ■ ADCCP ■ DDCMP ■ Bisync
- Analysis of a Specific Implementation

3. THE NETWORK LAYER

Design of the Network Layer
- Message Segmentation & Reassembly
- Virtual Circuits & Datagrams
- Routing (Path Selection) Techniques
- Priority, Flow, Error & Congestion Control

The X.25 Protocol
- ACK: Local and Remote Significance
- Control & Data Frame Mechanisms
- Implementations of Facilities/Options
- Source/Destination Addressing
- Reset & Restart of Virtual Calls

Implementing X.25: A Detailed Example
- Multiplexing of Virtual Circuits
- Flow Control on a Per-Circuit Basis
- Handling End-to-End Aspects of X.25
- Available VLSI Support Chips

Value-Added Networks (VAN's)
- Telenet ■ Tymnet ■ Teleglobe
- Transpac ■ Datapac ■ IPSS
- Euronet

4. THE TRANSPORT LAYER

Transport Layer Services
- Interprocess Communications Support
- Virtual Circuit Services
- When a Datagram Service is Preferred

Classes of Transport Protocols
- TCP: The High-Functionality DoD Protocol
- The Five ISO Classes of Transport Protocols
- The Two Classes of NBS Transport Protocols

Transport Layer Mechanisms
- Flow Control ■ Error Detection
- Byte vs. Block Sequence Numbers
- Resynchronization of Sequence Numbers
- Crash Recovery at the Transport Layer
- Opening and Closing Connections

5. THE HIGHER ISO LAYERS

ISO Session Layer Services
- Connection Management
- Synchronization Points
- Token and Interaction Management

Current Commercial Session Layer Services
- SNA's Bracketing & Chaining
- Name Server ■ Access Control

Evolving ISO Presentation Layer Services
- Negotiation of Transfer Syntax
- Abstract Representations
- Context Management

Current Commercial Presentation Services
- Virtual Terminal Protocols
- CCITT Parameterized Terminal Handling
- File Transfer Protocols
- Data Compression

Evolving ISO Application Layer Services
- Common Application Service Elements (CASE)
- Specific Application Service Elements
- File Transfer & Access ■ Job Transfer
- Message Handling ■ Virtual Terminals

Other Application Services
- Distributed Data Base Services
- Distributed Processing Services

6. GATEWAYS BETWEEN NETWORKS

"Translating" Gateways
- Protocol Conversion Functions
- Device Emulation

International Gateways
- X.75 Internetting ■ X.121 Addressing

Datagram Internets
- DoD Internet Protocol (IP)
- Internet Control Message Protocol (ICMP)
- Xerox PUP ■ ISO Internet Protocol

7. CONTROL, RELIABILITY & SECURITY

Network Monitoring & Control
- Usage Monitoring ■ Network Optimization
- Failure Determination and Isolation
- Automated Forms of Network Control
- ISO Network Management

Network Reliability & Availability
- Methods for Determining Reliability
- Improvements through Redundancy
- Measures of Availability

Network Security
- Network Vulnerability Analysis
- Data Encryption Standard vs. Public Key
- Electronic (Cryptographic) Signatures

8. PUTTING IT ALL TOGETHER

Protocol Suites
- IBM's SNA ■ DECnet
- X.400 Text/Graphics/FAX/Voice System
- MAP: Manufacturing Automation Protocol
- TOP: Technical Office Protocol

9. COURSE REVIEW & SUMMARY

INTEGRATED COMPUTER SYSTEMS

ENROLL NOW!

Call Toll Free **(800) 421-8166** or East Coast Office **(703) 847-6448** West Coast Office **(213) 417-8888**

Micro-Mainframe Connectivity: Practical Linking Tools & Techniques

8805 UB 256

DEMO LABS

Course 256



3 Days

OTTAWA
June 8-10
LOS ANGELES
July 13-15
TORONTO
July 13-15
WASHINGTON, D.C.
July 20-22
SAN DIEGO
August 10-12
LOS ANGELES
October 19-21
TORONTO
October 26-28
WASHINGTON, D.C.
October 26-28
OTTAWA
December 14-16

Courses at Your Facility
Please Inquire

Course Benefits

Sophisticated PC applications have created a growing need for Connectivity which allows PC users to access and share mainframe-based data. Micro-Mainframe Links satisfy this need by permitting the PC's capabilities to be amplified by the mainframe's established databases and processing power.

This course provides participants with the practical knowledge and skills required for the successful implementation of micro-mainframe links. **In-class PCs are used to demonstrate the key concepts and techniques presented.**

YOU WILL LEARN HOW TO:
- **Implement a Wide Variety of Connectivity Solutions**
- **Use Local and Remote Asynchronous Communication Links**
- **Conduct File Transfers with Full Error Checking**
- **Extract, Download and Reformat Mainframe Data**
- **Establish a Synchronous Connection to a Host Computer**
- **Evaluate Bisync and SNA Emulation Hardware and Software**
- **Utilize Peer-to-Peer Communication Tools**
- **Anticipate and Plan for Growth in Cooperative Processing**

WHO SHOULD ATTEND:
This course is valuable for MIS and telecommunication professionals, information center and office automation specialists, engineers, managers and end-users who are involved in the design, selection, implementation or use of micro-to-mainframe technology.

Course Content

1. MICRO–MAINFRAME CONNECTIVITY

Understanding Basic Concepts
- Categories of Micro–Mainframe Links
- Asynchronous and Synchronous Communications
- Critical Protocols and Standards
- System-Wide Database Integrity and Security

Data Communications Concepts
- Terminology: Parity, Duplex, Data Bits, Stop Bits, Switched and Leased Lines
- Data Terminal and Communication Equipment (DTE, DCE)
- Digital and Analog Transmission
- Parallel vs. Serial Transmission
- Error Handling and Synchronization Techniques
- Protocol Conversion

Capabilities of a Micro–Mainframe Link
- Terminal Emulation: Online to Mainframes
- File Transfer: Downloading/Uploading Data
- Selective Data Extraction and Transformation
- Cooperative Processing

System Network Architecture (SNA)
- Device Emulation
- Logical and Physical Units (LUs, PUs)
- Peer-to-Peer Processing: APPC (LU 6.2)
- Distributed Processing: SNADS

2. MAINFRAME CONNECTIVITY ISSUES

Understanding Host Data Structures
- Sequential and Indexed Files
- Hierarchical Database Systems
- Facilities of Fourth Generation Languages

Maintaining Data Integrity
- Insuring Responsibility for Data
- Aligning Host and Micro Data
- Audits and Audit Trails

Implementing Security
- Profiles and Passwords
- Protecting Existing Systems
- Physical Security
- Administrative Policies and Procedures

3. USING MICRO–MAINFRAME LINKS

In-Class Micro–Mainframe Labs
- Using Modems and Advanced Link Software
- Interacting With Host Computers
- Using Wide Area Nets: Telenet, Tymnet
- Reformatting Mainframe Data

Terminal Emulation Tools and Techniques
- Direct vs. Remote Communications
- Asynchronous Terminal Emulators: Blast, Crosstalk, SmartCom II, Relay, Others
- SNA and Bisync 3270 Emulation: AST, Cleo, Forte, IBM, IRMA, ITI, Micro Plus, NSA, PCOX, pcPath, Persyst, Others
- Multi-Session and Multi-Window Emulation
- 5250 Emulation: S/36, S/38

Performing File Transfers With PCs
- Screen Capture Versus Full File Transfer
- Tempus-Link, Omnilink, PC-Support
- Utilizing XMODEM, Kermit, MNP, X.PC
- Binary vs. Text File Transfer Implications
- 2780/3780 Remote Job Entry
- Data Compression Techniques

Departmental Mini Links
- Connecting Micros to Minicomputers
- Using Minicomputers as 3274 Gateways
- Virtual Storage for Microcomputers

Protocol Conversion
- Asynchronous to SNA
- Ethernet to SNA Bridges
- SNA to X.25 Gateways

4. ACQUISITION OF HOST DATA

Data Extraction and Reformatting
- Accessing Mainframe Data: IMS, VSAM, DB2, IDMS, FOCUS, ADABASE, NOMAD
- Selecting and Extracting Specific Data
- Transforming Host Data to PC Formats: WKS, CSV, DIF, SDF, PRN
- Database Centered Products
- Integrated Software: FOCUS, IFPS, SAS

5. EVALUATING AND SELECTING A LINK

Key Factors Influencing Link Selection
- Cost ■ Performance ■ Transfer Rate
- Integration with Existing Environment
- Support and Service

Survey of PC/Mainframe Link Software
- Micro/Answer ■ Infogate ■ Tempus-Access
- T-A-C ■ Golden Gate ■ ExpertLink
- Enhanced Connectivity Facilities
- FOCUS ■ ORACLE ■ Others

Cost Components of Micro–Mainframe Links
- Communication Hardware
- Intelligent Host Software
- Direct Access Storage Devices
- Interface Boards

6. ADVANCED TOPICS AND TRENDS

- Impact of the Token Ring LAN
- Gateways for LANs
- Program-to-Program Communications
- Server-Requestor Program Interface
- IBM Strategy: DB2, DISOSS, 9370 Line, DIA/DCA, SNADS
- LU 6.2 Products

DEMO LABS

In this course you will participate in a series of **demonstrations using an in-class PC** to:
- **Upload and Download Files**
- **Establish a PC-to-Host Session**
- **Extract and Reformat Host Data**
- **Link PC-to-Host via Public Networks**

Live and Video Demonstrations Include:
- **Asynchronous, 7-bit ASCII Transfers**
- **Synchronous, 8-bit Binary Transfers**
- **SNA Protocol and 3270/3274 Emulation**
- **Intelligent Link Demonstrations**



Processing downloaded mainframe data on a PC.

INTEGRATED COMPUTER SYSTEMS

ENROLL NOW!

Call Toll Free **(800) 421-8166** or

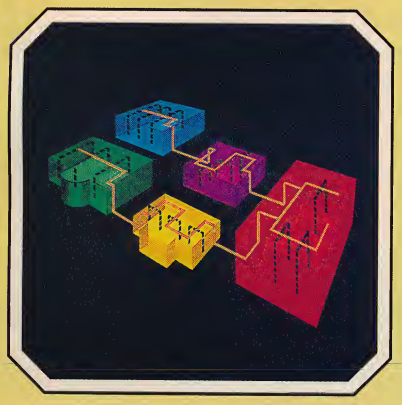
East Coast Office **(703) 847-6448**

West Coast Office **(213) 417-8888**

8805 UB 352

Local Area Networks: Implementation & Configuration

Course 352



4 Days

WASHINGTON, D.C.
May 24-27
PALO ALTO
June 14-17
WASHINGTON, D.C.
June 21-24
LOS ANGELES
July 19-22
WASHINGTON, D.C.
July 26-29
SAN DIEGO
August 23-26
WASHINGTON, D.C.
September 27-30
SAN FRANCISCO
October 18-21
BOSTON
October 25-28

Courses at Your Facility
Please Inquire

Course Benefits

The proliferation of powerful workstations and PCs, together with a vast installed base of minicomputers and mainframes, has produced **immense pressure** to link these resources and to increase productivity. Local Area Networks (LANs) allow the sharing of programs, data and peripherals by providing common access to local and remote resources.

This course provides participants with the fundamental knowledge needed to design, configure and implement LANs. **The course emphasizes the integration of available software and hardware elements, and provides a solid understanding of network architectures and protocols.**

YOU WILL LEARN HOW TO:
- **Understand Basic LAN Technology, Architecture & Protocols**
- **Evaluate Key HW/SW/Cost Tradeoffs for LAN Applications**
- **Select LAN Technology Based on Application Requirements**
- **Configure Local Area Networks to Interconnect the Computing Resources of Your Organization**
- **Implement Gateways to Other LANs and Wide-Area Nets**

WHO SHOULD ATTEND:

This course is valuable for systems analysts, design and development engineers, programmers, technical managers, and all data processing, telecommunications and office automation professionals & managers involved in designing, configuring or implementing Local Area Networks. Course 350, "Introduction to Datacomm & Networks" or equivalent background is recommended.

Course Content

1. INTRODUCTION & OVERVIEW

What is Local Area Network (LAN)?
- Definitions & Terminology
- Important Characteristics of LANs
- How LANs Differ from Wide Area Nets (WANs)
- Basic LAN Communications Concepts

Applications & Benefits
- Networking PCs, Workstations, Minis & Mainframes
- Resource Sharing: Disks, Printers, & High-Speed Processors
- Electronic Mail and Message Systems
- Internetwork Gateways to Remote Resources
- Engineering, Manufacturing & Office Automation Applications

2. FUNDAMENTALS OF LAN SYSTEMS

Basic LAN Approaches
- Bus Architectures & Cable Technology
- Baseband & Broadband Implementations
- Shared Media/Broadcast Systems
- Integrated Voice/Data Networks

Local Area Network Topologies
- Star ■ Ring ■ Tree
- Multi-Access Bus Techniques

Local Network Protocols
- Protocol Definitions & Terminology
- Functions of Layered Protocols
- Hardware, Software & Network Interfaces

3. LAN TECHNOLOGY & ARCHITECTURE

Local Network Media
- Shielded and Unshielded Twisted Pair
- Coaxial Cable: Broadband vs. Baseband
- Fiber Optic ■ Using the IBM Cabling System
- Performance, Cost & Installation Issues

Hardware Components
- Passive Cable Taps ■ Repeaters
- Active Ring Interfaces
- CATV Components
- Transceivers ■ Interface Processors
- Network Control Stations
- Interchannel Bridges
- Internetwork Gateways

Baseband Access Control Techniques
- Carrier Sense Multiple Access (CSMA)
- Collision Detection (CD) & Avoidance (CA)
- Slotted Rings ■ Token Passing

Broadband Approaches
- Frequency Allocation: Dedicated, Switched & Shared Channels
- Frequency Agile Modems
- CSMA ■ Token Bus
- Midsplit, Highsplit, & Dual Cable Systems

PABX-Based LAN Systems
- Integration of Voice & Data
- Use of Existing Telephone Wiring
- Reliability ■ Impact of ISDN

The Physical & Data Link Functions
- Packet Framing
- Contention Control & Back-off Algorithms
- Worst-Case Access Times
- Error Detection ■ Flow Control
- Logical Link Control

The Network Layer
- LAN/LAN, LAN/WAN Internetting
- Satellite Gateways to Remote LANs

4. EVALUATING & SELECTING LANS

Understanding & Applying the Standards
- The ISO/OSI Model, Relation to IEEE 802
- Domains of IEEE 802.2, 802.3, 802.4, 802.5
- MAP & TOP Standardization Activities

Key Factors Influencing LAN Selection
- Performance ■ Capacity
- Expandability ■ Cost-per-Port
- Installation & Cabling Considerations
- Reliability ■ Support & Service

Survey of LAN Products
- Baseband: Ethernet, NiNet, Net/One, Omninet, StarLan
- Broadband: Wangnet, LocalNet, Net/One
- Token Passing: IBM Zurich Ring, Token Bus
- Slotted Ring: Racal-Milgo (Cambridge Ring)
- PABX: Rolm, Intecom, ATT/IS, Northern Telecom, Mitel, Ericsson
- PC LANs: IBM PCnet & IBM Token Ring, Appletalk, Novell, 3Com, Others

Survey of LAN Software
- The TCP/IP Protocol Suite
- XNS Protocols ■ MAP/TOP Protocols
- Vendor Proprietary Protocol Software
- IBM's NETBIOS, APPC & LU 6.2 for LANs

5. CONFIGURING & OPERATING LANS

Configuring a LAN
- Determining Relevant User Needs
- Identifying Network Devices
- Predicting Traffic Load
- Analyzing Performance Requirements
- Specifying the Architecture & Components

Planning the Physical Layout
- Cable Layout & Routing Considerations
- Wire Closets for Fault Isolation/Recovery
- Installation Planning Guidelines

LAN Administration & Control
- Network Configuration Management
- Monitoring Traffic & Resource Utilization
- Automatic Fault Detection
- Hardware & Software for Network Security
- LAN Protocol Analyzers

6. COURSE REVIEW & SUMMARY

LAN Application Design Studies

During the course, participants will learn to develop detailed Local Area Network solutions through in-class design studies that cover both baseband and broadband applications.

Participants will learn how to:
- **Analyze an Application and Develop a Requirements Definition**
- **Determine the Network Architecture**
- **Select a LAN Technology and Specify the Required Components**
- **Determine Key LAN Features which Impact the Selection Process**
- **Develop Implementation & Network Management Guidelines**

INTEGRATED COMPUTER SYSTEMS

ENROLL NOW!

Call Toll Free **(800) 421-8166** or East Coast Office **(703) 847-6448** West Coast Office **(213) 417-8888**

8805 UB 253

PC Networking: Hands-On Workshop

HANDS-ON

Course 253



3 Days

OTTAWA
June 1-3
BOSTON
June 22-24
TORONTO
July 6-8
WASHINGTON, D.C.
July 13-15
LOS ANGELES
July 20-22
OTTAWA
August 17-19
SAN DIEGO
August 24-26
WASHINGTON, D.C.
September 14-16
PALO ALTO
September 28-30

Courses at Your Facility
Please Inquire

Course Benefits

The need for Connectivity and Resource Sharing to improve office productivity and transaction processing for millions of installed PCs is being met by the installation of a new generation of PC Networks. The rapid acceptance of the IBM Token-Ring LAN hardware, coupled with the software provided by DOS 3.1 and NetBIOS, has greatly enhanced the practicality of linking multiple PCs and peripherals into effective Local Area Networks.

The best way to learn about PC Networks is to USE ONE under the guidance of an expert instructor. This course provides the practical knowledge and HANDS-ON EXPERIENCE required to successfully install and use a PC network.

YOU WILL LEARN HOW TO:

- **Configure Network Hardware and Software**
- **Use the In-Class PC LAN to Run Multi-User Applications**
- **Optimize Network Performance and Provide Security**
- **Share and Control the Use of Peripherals, Software and Data**
- **Evaluate the IBM PC Net, Token Ring and Other Networks**
- **Choose an Appropriate LAN Operating System**
- **Effectively Utilize Gateways to Access Host Computers**

WHO SHOULD ATTEND:

This course is valuable for MIS professionals, telecommunications engineers, information center and office automation specialists, programmers, managers and end-users who are involved in the design, selection, implementation or application of PC LAN hardware and software technology.

Course Content

1. OVERVIEW OF IBM PC NETWORKS

Basic Concepts and Terminology
- Connectivity and PC Networks
- Sharing Resources, Applications and Data
- Star, Ring, Tree and Bus Topologies
- Broadband and Baseband Systems
- LANs vs. Multi-User Systems

Network Hardware
- Media: Twisted Pair, Coax, Fiber Optics
- The IBM Cabling System
- Servers and Workstations
- Channel Access: CSMA/CD, Token Passing
- Gateways and Bridges

Network Software
- Microsoft DOS Redirector (DOS 3.1)
- I/O Processing: Microsoft/IBM NetBIOS
- Network Operating Systems
- Multi-User Application Packages
- Network Standards and Protocols

2. HANDS-ON EXPERIENCE WITH PC NETS

Using the In-Class PC Network
- Bringing the Network Up
- Sharing Disks, Directories, Files, Software
- Electronic Messaging

Network Installation
- Configuring Servers and Workstations
- Organizing Directory Structures and Paths
- Establishing Security

Installing Application Software
- Application Setup and Initialization
- Multi-user (Network) Software Packages
- Handling "Non-Network-Ready" Software

3. LAN NETWORK HARDWARE

Hardware Components
- Network Adapter Cards
- Cabling and Connectors

Network Servers
- Print, Disk and File Servers
- Dedicated vs. Non-dedicated
- Communication Servers: RJE, SNA, X.25

Workstations
- Standard PCs and PC Clones
- Workstation/Server Combinations
- Diskless Workstations

Overview of Typical Network Products
- IBM Token Ring ■ Novell Netware
- G/Net ■ StarLAN ■ ProNET-10 ■ ARCnet
- Banyan ■ Ethernet ■ Tapestry ■ Others
- Serial Port LANs: EasyLAN, IDEAshare, LANLink

4. PC NETWORK SOFTWARE

The IBM Token Ring PC Network
- DOS 3.1 (or higher) and NetBIOS
- The Network Operating System: MS-NET
- NetView/PC ■ Multistation Access Unit
- Repeaters and Bridges

The Novell Approach
- NETWARE: Multitasking LAN Operating System
- Disk Caching and Elevator Seeking
- System Fault Tolerance (SFT)
- Third Party Network Support: 3COM, IBM Token Ring, AT&T StarLAN, Proteon, Others
- Bridging Dissimilar Networks

Multi-User Application Software Examples
- Microsoft Word ■ Multimate ■ Wordperfect
- Lotus 1-2-3 ■ Symphony ■ SuperCalc
- R:base 5000 ■ dBASE III Plus ■ Revelation

5. REMOTE COMMUNICATIONS

Communicating with Modems
- Crosstalk XVI, Others
- Workstation vs. Network Modems

Connecting LANs to Host Computers
- 3274 Controller Emulation
- Multi-Session Emulation
- File Transfer Approaches

Gateways and Bridges
- SNA, Asynch and X.25 Gateways
- Ethernet to Token Ring Interconnection
- X.25 Public Network Support

6. SELECTING AND CONFIGURING A LAN

Key Factors in LAN Selection
- Architecture, Speed and Cabling
- Product Support and Service
- NetBIOS Support
- SNA (LU 6.2) Integration
- Host Connectivity ■ IEEE Standards

Pitfalls to Avoid
- Integration with Existing Environment
- Media/SW Cost-Performance Tradeoffs
- Expandability and Cost per Station

7. LAN ADMINISTRATION

LAN Administration and Control
- Assigning User IDs and Passwords
- Controlling and Allocating Resources
- System Documentation
- Special Purpose Batch Files

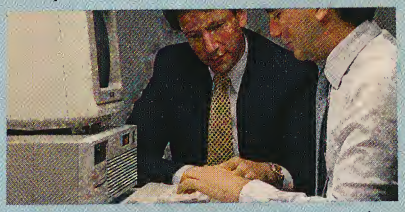
Network Management Guidelines
- Version Control and Software Update
- Controlling the User Environment
- IBM's Netview and Netview-PC
- Tuning the Network's Performance
- Back-up and Recovery Procedures

HANDS-ON

Throughout the course, **In-Class PCs are used to provide Extensive Hands-On PC Network experience**, supplemented by video and other media presentations.

Hands-on exercises include:

- **Installing and Tuning an IBM Token Ring Network**
- **Employing Network Commands**
- **Executing Standard Packages**
- **Running Multi-User Software**
- **Using the Novell Advanced NetWare Operating System**



Participants use an In-Class PC as a workstation on the IBM Token Ring Network.

INTEGRATED COMPUTER SYSTEMS

ENROLL NOW!

Call Toll Free **(800) 421-8166** or

East Coast Office **(703) 847-6448**

West Coast Office **(213) 417-8888**

MAP/TOP: Technical Introduction & Implementation Status Report

Course 357



4 Days

BOSTON
May 24-27
TORONTO
June 7-10
WASHINGTON, D.C.
July 26-29
LOS ANGELES
August 30-September 2
MONTREAL
September 13-16
BOSTON
September 27-30
TORONTO
November 29-December 2
WASHINGTON, D.C.
December 6-9

Courses at Your Facility
Please Inquire

8805 UB 357

Course Benefits

MAP, the Manufacturing Automation Protocol, and TOP, the Technical and Office Protocols, provide the only multi-vendor local area networks available today based on international standards (Open Systems Interconnection) not subject to the vendors varying implementations.

This course provides a thorough introduction to the data communications and computer network technology defined by MAP/TOP. The emphasis is on the critical trade-offs which must be made by implementors and providers, and how those trade-offs impact application developers and end users.

YOU WILL LEARN:

- **MAP/TOP Network Technology, Architecture & Protocols**
- **How MAP/TOP Data Communication Hardware and Software Work, from an Engineering Point of View**
- **Current Status of MAP/TOP Standards & Vendor Products**
- **Guidelines for Evaluating Benefits of MAP/TOP Versus Alternative Networking Approaches**
- **How to Deal Successfully with the Key Strategic and Tactical Issues Involved in Applying MAP/TOP Technology**
- **Migrating from Current Networks to Open Systems Networks**

WHO SHOULD ATTEND:

This course is valuable for technical professionals and managers responsible for implementing network solutions using multi-vendor computer network components. MAP and TOP are the leading approach to the application of OSI for general networking needs. However, the issues raised in this course also apply to networks based on other protocols.

Course Content

1. INTRODUCTION & OVERVIEW

Basic Concepts
- What are MAP and TOP?
- Computer Integrated Manufacturing (CIM) & Offices (CIO): Strategies & Tactics
- Communication & Flexible Automation
- Automation of Design and Engineering
- Linking "Islands of Automation"
- Opportunities & Challenges

Evolution of MAP/TOP Networks
- The Role of General Motors, Boeing and the National Bureau of Standards
- How MAP, TOP & GOSIP Work Together
- Support from: IBM, DEC, HP, Others
- MAP/TOP User Groups Around the World
- Other Groups Supporting MAP/TOP

Features & Benefits of MAP/TOP
- Interoperability & Vendor Independence
- Application Portability & Flexibility
- Reduced Customization
- Enhanced Productivity & Quality
- Facilitation of Flexible Automation
- Reduced Cost of Network SW & HW

Drawbacks & Liabilities of MAP/TOP
- Cost and Availability Today
- Performance and Complexity
- Conformance & Interoperability Testing
- Vendor Commitment and User Commitment

2. FUNDAMENTALS OF MAP/TOP NETWORKS

Communication Requirements
- Industrial, Office, and Engineering
- Local, Metropolitan & Wide Area Networks
- Interconnecting Multi-Vendor Equipment
- Reliability & Fault Tolerance
- Real-Time/Time-Critical Applications

Understanding Network Layers & Protocols
- The ISO/OSI Network Reference Model
- Protocols vs. Interfaces vs. Services
- MAP/TOP Networks & the OSI Model
- How Other Standards Fit In: CCITT, ANSI, ECMA, EIA, IEEE, ISA, NBS

Basic MAP/TOP Network Concepts
- Baseband, Carrierband, Broadband
- Coax, Fiber Optic, Twisted Pair
- Nodes, Links, Packets
- Token Bus, Token Ring, CSMA/CD
- Backbone, Departmental & Cell Networks
- Routers, Bridges, & Gateways
- MiniMAP and Enhanced Performance Architecture

3. THE PHYSICAL, DATA LINK & NETWORK LAYERS

The MAP Physical Layer
- The Broadband, 10 Mbit/s RF Connection
- Standard CATV Parts
- Key Aspects of RF Modems & Interfaces
- Single-Channel Carrierband Modems
- The Token Passing Bus Protocol
- Message Prioritization Capabilities

TOP Physical Layer Options
- Cabling & Baseband Signaling
- IEEE 802.3 CSMA/CD Design
- Thick Coax, Thin Coax & Twisted Pair
- IBM Token Ring (IEEE 802.5)

The MAP/TOP Data Link Layer
- MAP/TOP Upper Level Compatibility
- IEEE 802.2 Logical Link Control
- Connectionless Service
- Single Frame Service

The MAP/TOP Network Layer
- Connectionless Interconnect Protocol
- Routing Packets: Inter- & Intra-Network
- Internetworking Architectures
- Connectivity to X.25 Wide Area Nets
- MAP/TOP Network Addressing Conventions

4. MAP/TOP TRANSPORT, SESSION & PRESENTATION LAYERS

The Transport Layer
- END-to-END Data Quality Control
- Data Packet Assembly & Disassembly
- Error Detection & Recovery
- Flow Control
- Transport Classes

The Session Layer
- Session Functional Units
- Establishing Application Dialogs
- Synchronization and Rollback
- Negotiated Release

The Presentation Layer
- MAP 2.1 vs MAP/TOP 3.0
- Defined Context Sets
- Abstract Syntax Notation One (ASN.1)
- Abstract and Concrete Encodings

5. MAP/TOP APPLICATION LAYER SERVICES

Manufacturing Oriented Services
- Manufacturing Messaging Format Standard (MMFS)
- Manufacturing Messaging Standard (MMS)
- MMFS to MMS Migration
- Initial Graphics Exchange Standard
- Product Data Exchange Standard

Office Oriented Services
- X.400 Message Handling Service
- Virtual Terminal Service
- Office Document Architecture
- Office Document Interchange Formats
- Graphics Kernel System
- Computer Graphic Metafile

Common Services
- File Transfer, Access and Management (FTAM)
- The Network-Wide Directory Database
- Network Management Support
- Association Control Service Elements

6. IMPLEMENTATION CONSIDERATIONS

MAP/TOP Products & Vendors
- Chip, Board & Stand-Alone Systems
- Starter Kits and Component Products
- Cable Plant and Modem Considerations
- Vendor Commitments and Deliveries
- The MAP/TOP Product Directory

Prerequisites to Success
- Designing the Network Configuration
- Cable Plant Selection & Interconnection
- Developing an Implementation Plan
- Installation & Start-up Considerations
- Testing Conformance & Interoperability
- Network Management Requirements & Tools
- Education, Documentation & Maintenance

Migration Plans and Strategy
- Pilot Projects to Production Use
- MAP 2.1 to MAP/TOP 3.0
- Proprietary Networks to MAP/TOP
- TCP/IP to TOP/GOSIP
- TOP 3.0 to future TOPs

MAP/TOP Case Studies

7. ADVANCED TOPICS & ENHANCEMENTS

- Impact of ISDN & Wide Area Net Services
- Fiber Data Distributed Interface (FDDI)
- Artificial Intelligence & Expert Systems
- Evolution of CIM & Office Applications
- Evolution of OSI Services

INTEGRATED COMPUTER SYSTEMS

ENROLL NOW!

Call Toll Free **(800) 421-8166** or

East Coast Office **(703) 847-6448**

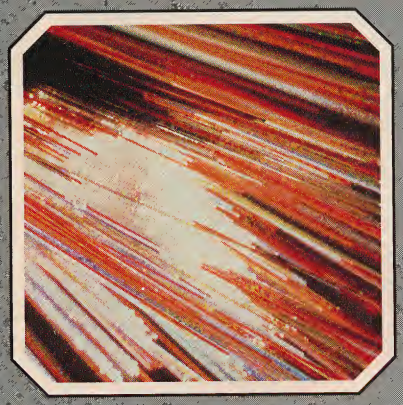
West Coast Office **(213) 417-8888**

Photo courtesy of Siecor Corporation.

8805 UB 440

Fiber Optic Communication Systems: Design Tools & Techniques

Course 440



4 Days

BOSTON
May 24-27
WASHINGTON, D.C.
June 14-17
SAN FRANCISCO
June 21-24
LOS ANGELES
July 12-15
SAN DIEGO
August 2-5
WASHINGTON, D.C.
August 16-19
BOSTON
September 13-16
SAN FRANCISCO
September 27-30

Courses at Your Facility
Please Inquire

Course Benefits

Fiber Optic cable has emerged as the "media of choice" for high-capacity voice and data communications. Successful, cost effective applications range from long-haul telecommunications to intra-facility data links.

This course provides BOTH the fundamental principles AND the practical engineering tools and techniques for fiber optic system specification, design, and implementation.

YOU WILL LEARN HOW TO:

- **Understand Electro-Optic Components and their Behavior**
- **Specify & Design Point-to-Point and Multi-Hop Transmission Systems**
- **Evaluate and Select Fiber Optic System Components Including: Cable, Sources, Detectors & Connectors**
- **Develop the Design of the Transmitter & Receiver Systems**

WHO SHOULD ATTEND:

This course is valuable for system and design engineers, technical managers, technicians and others who are responsible for specifying, designing, or maintaining Fiber Optic Systems.

TAKE-HOME FIBER OPTIC SYSTEM DESIGN SOFTWARE

Participants receive an IBM-PC diskette with 1-2-3/Symphony models for Transmission System Design and Single Mode Systems. These models permit the user to determine Receiver Power, Signal/Noise, Loss, Rise-Time, Bandwidth and other key performance parameters of fiber optic communication systems.

Course Content

1. INTRODUCTION TO FIBER OPTICS (FO)

Basic Fiber Optics Concepts
- Terminology and Jargon
- The Evolution of Fiber Optic Systems
- Transmission of Light in Optical Fibers
- Light Sources and Detector Technology
- Design of a Typical Data Link

Features and Benefits of Fiber Optics
- Comparison with Hard-Wire and Microwave
- Built-In, Low-Cost Growth Capacity
- Small Size ■ Wide Bandwidth
- Reliability ■ EMI Immunity

Steps in Fiber Optics Data Link Design
- Understanding the Requirements
- Resolving Trade-Offs in Distance, Data Rate, and Interfaces
- Choosing an Appropriate Cable
- Specifying the Source/Detector Elements
- Checking the Overall System Design

Overview of Fiber Optics Applications
- Telecommunications: Long Haul, Metropolitan Trunks & Subscriber Loops
- On Premise/"Campus" Data Links
- Military/Aerospace Systems
- Office/Industrial Local Area Networks
- Medical & Scientific Applications

2. FIBER OPTIC TECHNOLOGY

Basic Principles of Electro-Optics
- The Nature of Light and its Propagation
- Measurement of Refraction and Absorbtion
- Effects of Coherence and Polarization
- Applications of Lenses, Mirrors, Beam-splitters, and Filters

Fiber Optic Cable
- Fiber Types and How They Differ: Single Mode, Graded and Stepped Index
- Understanding Key Cable Specifications: Index of Refraction, Numerical Aperture, Scattering and Reflection
- Information Transfer Limitations: Attenuation, Distortion, and Bandwidth
- Splice and Connector Considerations

Signal Sources
- LASERs and Light Emitting Diodes (LEDs)
- Cost/Performance Trade-Offs
- Surface vs. Edge Emitting LEDs
- Wavelength, Bandwidth & Power Level
- Modulation and Stabilization Techniques

Signal Detectors
- Photodiodes: PIN and Avalanche (APD)
- Equivalent Power & Responsivity
- Calculating Response Time & Noise

Coupling Fibers to Emitters and Detectors
- Packaging: DIP with Pigtails, TO Cans, Hybrid Chips, and Panel Mounts
- Handling and Mounting Considerations
- Matching Numerical Aperture
- Alignment: Radial, Angular, and Separation

3. TOOLS FOR DEVELOPMENT & TEST

Optical Time Domain Reflectometer
- Interpreting Cable Profiles
- Determining Cable and Link Attenuation
- Detecting and Locating Cable Damage

Power, Pulse, and Spectrum Measurements
- Performing Basic Power Measurements
- Determining Splice and Connector Losses
- Measuring Distortion & Modulation Depth
- Measuring Signal Spectrums and SNR

Measuring Bit Error Rate and Link Margins
- The Relationship between SNR and BER
- BER Estimation and BER Testers
- Calculating Operating Margins

4. IMPLEMENTING FIBER DATA LINKS

Cable Selection Considerations
- The Distance vs. Bandwidth Tradeoff
- Low Cost/Short Haul Cables
- High Performance Cables

Transmitter/Receiver Circuit Design
- Low-Cost, On-Board Systems
- Systems for Mid-Range Applications
- Maximum Performance Designs
- Modulation and Encoding Techniques
- Detection and Gain Control Schemes

System Level Design and Analysis
- Use of Fibers in Local Area Networks
- Properties of Multi-Hop Links and Repeaters
- Estimating the Power Profile (Flux Budget)
- Calculating Bandwith and Rise Time
- Designing Bit-Error-Rate Margins

Off-the-Shelf Systems and Components
- Board Level Transmitters and Receivers
- Low & High Speed Data Link Modules
- Multiplexed and Networked Data Links
- Couplers, Switches and Isolators

Implementation & System Testing
- Installation Planning
- Component Selection
- Pre-Installation Testing
- Benchmarking & Proof of Performance
- Periodic Inspection & Maintenance

5. SURVEY OF PRODUCTS & VENDORS

- Fibers and Cables
- Sources and Detectors
- Systems and Subsystems
- Test and Development Equipment

Fiber Optic Components
- Fibers and Cables
- Sources & Detectors

Development Tools & Systems
- Test & Development Equipment
- Systems & Subsystems

6. SPECIAL TOPICS AND FUTURE TRENDS

- Datacomm Network Systems
- Telephone T-Carrier Systems
- Fiber Digital Data Interface (FDDI)
- Emergence of Single Mode Systems
- Developments in Integrated Optics
- Coherent Systems ■ Video Systems

CASE STUDIES

Through detailed in-class case studies, participants gain experience in:

- **Developing the Specification for a Long-Haul, High Bandwidth Link**
- **Designing an Intra-Facility Data Link**
- **Designing a PC-Board Interconnect Link**

In addition, each concept is reinforced with in-class design exercises including

- **Calculating Optical Power Density**
- **Relating Dispersion to Bandwidth**
- **Computing Attenuation & Loss**
- **Predicting Signal-to-Noise Ratio**
- **Assessing Photodetector Dynamic Range**
- **Forecasting a Link's Bit-Error Rate**

INTEGRATED COMPUTER SYSTEMS

ENROLL NOW!

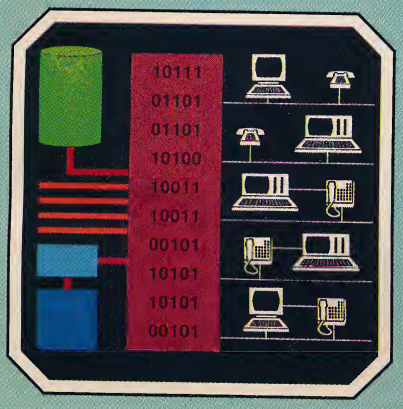
Call Toll Free **(800) 421-8166** or

East Coast Office **(703) 847-6448**

West Coast Office **(213) 417-8888**

Integrated Voice/Data Communications & ISDN

Course 354



4 Days

WASHINGTON, D.C.
June 14-17

LOS ANGELES
June 21-24

TORONTO
July 19-22

SAN DIEGO
July 26-29

WASHINGTON, D.C.
August 16-19

PALO ALTO
September 13-16

BOSTON
October 18-21

LOS ANGELES
October 25-28

WASHINGTON, D.C.
November 15-18

Courses at Your Facility
Please Inquire

Course Benefits

8805 UB 354

A new generation of Integrated Voice/Data equipment is being produced and installed today, providing services that improve productivity while reducing costs. The modern digital PABX and the evolution of the Integrated Services Digital Network (ISDN) represent a "merger" of voice and data communication technology. These and related devices and services are proving to be practical, cost-effective tools for improving the utilization and management of organizational communications.

This course provides a comprehensive background in the state-of-the-art of modern voice and data communications. **Emphasis is on the practical aspects of selecting, configuring, installing and maintaining an integrated voice/data system,** as well as on understanding the technologies that support that system.

YOU WILL LEARN HOW TO:

- **Evaluate the Benefits, Opportunities, Costs and Impacts of Modern Integrated Voice/Data Systems & ISDN**
- **Select a PABX Which Meets Current and Future Needs**
- **Differentiate the Essential Characteristics of Voice vs. Digital Data Communication**
- **Configure Networks for Intra-Facility, "Campus" and Long-Haul Communications**
- **Understand & Plan for the Evolution of ISDN**

WHO SHOULD ATTEND:

This introductory course is valuable for telecommunications, data processing, engineering and office automation professionals who need to broaden their knowledge of voice communication systems and improve their understanding of data communication principles and capabilities.

Course Content

1. INTRODUCTION & OVERVIEW

Basic Concepts
- Existing Public & Private Voice Networks
- Growth of Data Communications
- The Modern PABX "Switch"

Voice/Data Integration
- All-Digital Communication
- ISDN: Integrated Services Digital Network
- Merging Voice & Data Applications

Integrated Voice/Data Components
- Key Voice & Data Cost/Performance Issues
- Digital PABXs & Local Area Networks (LANs)
- Simultaneous Voice/Data Workstations

2. FUNDAMENTALS OF VOICE COMMUNICATION

Communication Fundamentals
- Bits, Bytes, Bauds & Bandwidth
- Signals, Coding, Transmission & Media
- Basic Switching Techniques

Transmitting Voice Signals
- Bandwidth & Channel Capacity
- Attenuation, Noise & Distortion
- Transmission Media

The Telephone Network
- Telephone Network Architecture
- Voice Station Equipment
- Trunks & Central Office Switching Systems
- Mechanical vs. Electronic Switching
- Local Loop & Toll Transmissions
- Voice Services

3. DIGITAL DATA COMMUNICATION

Digital Encoding Techniques
- Digital Voice Encoding
- PAM ■ PCM ■ ADPCM
- Companding ■ VLSI Devices: Codecs

Digital Data Transmission
- The DS1, DS2, DS3, DS4 Hierarchy
- Half & Full Duplex ■ Sync & Async

Digital Communication Equipment
- Voice/Data Modems
- Interfaces: RS-232C/RS-449; V.35
- Time, Frequency & Statistical Multiplexors

Switching Digital Data
- Basic Digital Switching Techniques
- Circuit Switching vs. Packet Switching

4. THE DIGITAL PABX

PABX Architectures
- Centralized vs. Distributed Control
- Accessability, Modularity & Redundancy
- Stored Program Control
- Non-Blocking & Multi-Stage Switching
- PABX-Based LANs ■ Stars, Rings & Tokens

Digital PABX Services
- Voice & Data Switching
- Least Cost Routing Algorithms
- T1 Interfaces & X.25 Pads
- Computer-to-PABX Interfacing (CPI, DMI)
- Protocol Conversion & Translation
- Store & Forward Voice Messaging
- Electronic Mail
- Teleconference Services

Integrated Voice/Data Systems
- Transmission & Switching Systems
- Communications Servers
- Cluster Controllers

Wiring Buildings for Voice and Data
- Broadband vs. Twisted Pair Cabling
- Wiring Plans: IBM, AT&T, NT, DEC

5. ISDN: FUNCTIONS & EVOLUTION

Applications and Opportunities
- Basic Service Philosophy
- Determining Economically Viable Services
- Centrex Extensions
- Portable Teleconference Services
- WATS Extensions

Structure, Layers & Access
- Network Termination: NT1 & NT2 Functions
- Basic User Network Interface (2B + D)
- U Interface
- Basic & Primary Access
- Link Access Protocol (LAP-D)

Support Systems & Hardware
- Common Channel Interoffice Signalling
- Signalling System No. 7
- VLSI Parts for Lower Layers

Status & Expected Developments
- North American & European Trials
- Terminal Equipment for ISDN
- Telco Medium-Term Plans

The PABX/Centrex/ISDN Debate
- Life-Cycle Costs
- User Control
- Availability of Features

6. WIDE AREA NETWORKS

Voice Carriers & Services
- Software Defined Networks
- LATAs ■ Other Common Carriers (OCCs)
- GTE ■ MCI ■ SBS
- Local Access Bypass Technology

Local & Long Distance Digital Services
- Dedicated Digital Facilities
- Switched Digital Facilities
- Digital Cross-Connect Services

Newer Services
- Cellular Mobile Telephone Service
- Fiber Optic Systems ■ Videotex
- Electronic Mail: INTELPOST

7. VOICE/DATA VENDORS & PRODUCTS

Modems & Multiplexors
- Codex ■ Timeplex

Overview of Digital PABXs
- AT&T ■ IBM/ROLM ■ GEC ■ Mitel
- Northern Telecom ■ Plessey
- GTE ■ Philips ■ Ericsson

Integrated Voice/Data Workstations

LAN Hardware & Software

8. IMPLEMENTATION CONSIDERATIONS

PABX Implementation Strategies
- Requirements Definition & Analysis
- Estimating Voice & Data Traffic
- Functional & Administrative Features
- Expansion, Upgrade & Maintenance
- Installation and Life-Cycle Costs

Managing Communications Services
- Monitoring & Measuring Traffic
- Reports: Call Detail: CDR & SMDR
- Automatic Line Quality Testing
- Self/Remote Diagnostic Services

Application Workshops
- Integrating Intra-Company Voice & Data
- Linking Regional & Branch Offices
- World-Wide Voice/Data Communications

Application Workshops

In-class Application Workshops will include:
- **Integrated Wide Area Network Costs**
- **Trunk Group Capacity**
- **PABX Requirements Definition**

INTEGRATED COMPUTER SYSTEMS

ENROLL NOW!

Call Toll Free **(800) 421-8166** or

East Coast Office **(703) 847-6448**

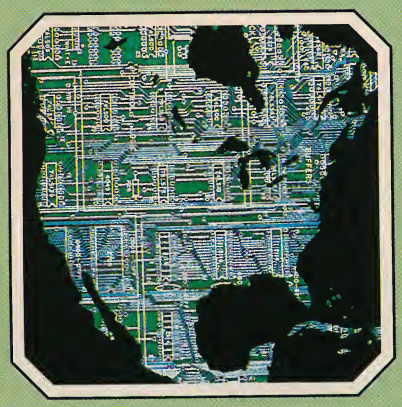
West Coast Office **(213) 417-8888**

UB

8805 UB 358

Digital Communication Systems: Design & Applications

Course 358



4 Days

SAN DIEGO
June 7-10
OTTAWA
June 14-17
LOS ANGELES
July 26-29
WASHINGTON, D.C.
August 2-5
SAN FRANCISCO
September 13-16
LOS ANGELES
October 18-21
OTTAWA
November 15-18
WASHINGTON, D.C.
January 17-20

Courses at Your Facility
Please Inquire

Course Benefits

The explosive growth of long haul and local computer networks, together with increased demands for secure military communication channels and error-free command and control systems, have fostered the rapid advance of digital data communication technology.

This course presents the fundamental concepts and design techniques of state-of-the-art digital communication systems. The emphasis of the course is on a careful analysis of individual data communication elements and the engineering synthesis of these elements into an effective system.

YOU WILL LEARN HOW TO:

- **Design Communication Links for Data, Voice, and Video**
- **Analyze and Model System Performance**
- **Apply Signal Quantization and Source Encoding Techniques**
- **Implement Binary, M-ary, and Hybrid Modulation/Demodulation**
- **Establish Error Control in Digital Data Systems**
- **Employ Data Encryption for Secure Communications**
- **Evaluate and Select Digital Communications Hardware**

WHO SHOULD ATTEND:

This course is valuable for design engineers, systems analysts, technicians, network designers and technical/project managers who are involved in the specification, design, or implementation of data communication systems. An engineering or technical background is assumed.

Course Content

1. INTRODUCTION & OVERVIEW

Data Communications & Computers
- Digital vs. Analog Communications
- Elements of Digital Systems
- Computer Network Requirements
- Fundamental Design Strategies

Data Communication Channels
- Voice Grade Lines
- Broadband & Baseband Cables
- Microwave & Satellite Transmission

2. DIGITAL SIGNALS & NOISE

Time Domain Characteristics
- NRZ-L, M, & S Signal Formats
- Bi-Phase-L, M, & S Signal Formats

Frequency Domain Characteristics
- Power Spectral Densities, Random & Non-Random Data
- Calculation & Measurement Techniques

Noise Characteristics
- Additive White Gaussian Noise (AWGN)
- Bandlimited AWGN Characteristics
- Other Noise Types

Optimum Receiver Design
- Matched Filter Detection
- Correlation Detection
- Receiver Synchronization

3. DATA SOURCE ENCODING TECHNIQUES

Waveform Digitization Techniques
- Pulse Code Modulation (PCM)
- Companded PCM
- Differential PCM (DPCM)
- Delta Modulation (DM)
- Adaptive DM (ADM & CVSD)

Data Compression Techniques
- Sample Prediction & Interpolation
- Run-Length Coding
- Other Redundancy Removal Techniques

Special Techniques for Voice & Video

4. MULTIPLEXING/DEMULTIPLEXING

Frequency Division Multiplexing (FDM)
- Advantages/Disadvantages
- Performance ■ Efficiency

Time Division Multiplexing (TDM)
- Synchronous Multiplexing
- Asynchronous Multiplexing
- Frame Structures
- Sub/Super Commutation

Frame Synchronization/Decommutation
- Search/Lock Strategies
- Linear Recursive Sequences, Barker, Willard, Neuman-Hofman Codes
- Frame Synchronizer Threshold
- Role in System Performance Monitoring

5. MODEMS: MODULATION & DEMODULATION

Binary Modulation
- Phase, Amplitude & Frequency Shift Keying
- Spectral Properties & Required Bandwidths
- Signal Space Representations

Binary Demodulation
- Coherent ■ Noncoherent
- Bit/Symbol Synchronization
- Performance Comparisons
- Implementation & Operation

"M-ary" Modulation/Demodulation
- M-ary Phase/Amplitude/Frequency
- Hybrid Techniques
- Power vs. Bandwidth Tradeoffs

6. DATA TRANSMISSION ERROR CONTROL

Error Detection Coding
- Basic Concepts, Capabilities/Limitations
- Bit, Symbol & Word Error Probabilities
- Throughput Calculations & Tradeoffs
- Applications-Oriented Code Selection
- Practical Implementation Techniques

Error Correction Coding
- Block & Convolutional Coding
- Decoding for Block & Convolutional Codes
- Hamming Single & Double Error Correction
- BCH Code & Performance Calculations
- Typical Implementation Approaches

7. DATA ENCRYPTION TECHNIQUES

Fundamental Principles & Applications
- Enhancement of Bit Synchronization
- Carrier Synchronization Enhancement
- Encryption vs. Authentication

Encryption/Decryption Approaches
- Linear Feedback-Shift-Registers
- Nonlinear Shift Registers
- Data Encryption Standard (DES)
- Public Key Cryptosystems
- Decryptor Error Extension Effects

8. COMPUTER COMMUNICATIONS LINKS

Computer Network Architecture
- Network Topologies
- The Public Telephone Network
- Data Traffic Requirements
- Packet, Message & Circuit Switching
- Local Area Network (LAN) Implementations

Communication Protocols
- The ISO-OSI Layered Model
- Protocol Functions at Each Layer
- X.21, X.25, SNA, ARPANET, DECNET

9. SYSTEM PERFORMANCE MEASURES

Network Link Calculations
- Techniques for Measuring Link Parameters
- Sources of Loss & Degradation
- Calculating Bit Error Rates

Monitoring System Performance
- Link Parameters
- Signal-to-Noise Ratios
- Synchronization Status

10. APPLICATIONS DEVELOPMENT & FUTURE TRENDS

Selection of Hardware Components
- Modems ■ Codecs
- Mux/Demux Equipment
- Bit/Symbol Synchronizers
- Encryptors/Decryptors
- Source Encoding Devices & Equipment

Digital Communication in Local Networks
- Cables ■ Fiber Optics
- LAN Communication Architecture
- Carrier Sense & Ring Networks

Application Examples
- Military Command and Control
- Data Acquisition & Tracking Systems
- Transmission via Telephone Circuits
- Digital Audio Recording Systems
- Ultra-High Data Rate Applications

11. COURSE REVIEW & SUMMARY

INTEGRATED COMPUTER SYSTEMS

ENROLL NOW!

Call Toll Free **(800) 421-8166** or

East Coast Office **(703) 847-6448**

West Coast Office **(213) 417-8888**

COURSE AUTHORS AND INSTRUCTORS*

■ Knowledge-Based Systems & Artificial Intelligence p. 2

■ Expert Systems Design Workshop p. 3

Dr. George Luger has been active in the development of AI techniques for over 10 years. He has developed and applied knowledge-based systems to a wide range of problem areas, including robot designs, production systems, and automated layout of integrated circuits.

Other Faculty: Dr. R. Adar, Mr. J. August, Dr. D. Barkley, Mr. A. Christiansen, Mr. G. Davidson, Dr. J. Durkin, Dr. G. Lazarev, Dr. G. Medioni, Mr. M. Mendelovici, Mr. M. Sims, Mr. W. Stubblefield, Mr. A. Thompson and Mr. B. Zack.

■ Introduction to Datacomm & Networks p. 4

■ Computer Network Design & Protocols p. 7

■ Local Area Network Implementation p. 9

Dr. Gerald Cole has 15 years of practical engineering experience in applying data communications technology to the design and development of computer networks. His implementation background includes multi-protocol front end processors, long haul, packet switched networks and the detailed design of hardware and software for high speed local area network products. Dr. Cole has served on national data communication standards committees, and is currently system architect for a state-of-the-art, secure military network.

Other Faculty: Mr. D. Beattie, Mr. R. Causey, Mr. J. DeMarco, Dr. A. Fortino, Mr. J. Gaspar, Mr. S. Johnson, Dr. V. Jones, Mr. J. Katz, Mr. A. Ma, Mr. L. Nidus, Mr. J. Passafiume, Mr. W. Roehr, Mr. M. Rothberg and Mr. K. Siyan.

■ Troubleshooting Datacomm & Networks p. 5

Mr. Donald Beattie has over 20 years of experience in the areas of telecommunications, switching systems and networks. His application background includes the design of telephone switches, digital interfaces, high speed packet networks, hardware and software for network monitoring, and a variety of micro/mini-to-mainframe links. Mr. Beattie lectures internationally and consults for large organizations in the area of voice and data communications, networks and troubleshooting.

Other Faculty: Mr. E. Braun, Mr. D. Freberg, Mr. J. McDermott and Mr. P. Smith.

■ Network Management & Control p. 6

Mr. Neil Hollingum has over 16 years of network management experience with large scale government and military computer networks. His practical implementation experience includes local area networks, long haul communications, packet switching, telephone systems and integrated voice/data networks. As an expert in the management and control of networks he has lectured internationally and consults in Britain, Europe, Scandinavia and the U.S.A.

Other Faculty: Mr. D. Beattie, Mr. J. Passafiume and Mr. J. Scott.

■ Micro–Mainframe Connectivity p. 8

■ PC Networking: Hands-On Workshop p. 10

Mr. Dwayne Walker has a broad background in Micro-to-Mainframe links and the networking of industrial and commercial PC workstations. His practical experience includes the implementation of the IBM, Novell, Banyan and 3COM PC networks for a variety of office automation, laboratory and industrial applications. He has developed PC links to IBM, DEC and HP mainframes, and is author of the popular text "Micro to Mainframe."

Other Faculty: Mr. T. Booth, Mr. M. deGuzman, Mr. J. Hurault, Mr. A. Ma, Mr. A. McGrath, Mr. J. Ramsey and Mr. K. Siyan.

■ MAP/TOP: Introduction & Status p. 11

Dr. Vincent Jones has been applying computer and network technology to the solution of practical problems for over 15 years. His experience ranges from the analysis of data handling for a worldwide packet switching network to the development of robot control systems based on artificial intelligence. He is author of the book "MAP/TOP Networking: A Foundation for Computer-Integrated Manufacturing."

Other Faculty: Mr. S. Johnson, Mr. J. Passafiume, Dr. M. Quinn and Mr. M. Rothberg.

■ Fiber Optic Communication Systems p. 12

Mr. Julius (Pat) Darden has worked for 25 years in military and commercial computer and communication systems. For the past six years he has specialized in the electronic and fiber optic system design and implementation of voice and data communication applications.

Other Faculty: Dr. J. Courtney-Pratt, Dr. M. Drake, Dr. M. Kale, Dr. A. Nelson, Mr. J. Scott, Mr. R. Verkon and Mr. P. White III.

■ Integrated Voice/Data Communications p. 13

Mr. Donald Beattie has over 20 years of experience in the areas of telecommunications, switching systems and networks. His application background includes the design of telephone switches, digital interfaces, high speed packet networks, hardware and software for network monitoring, and a variety of micro/mini-to-mainframe links. Mr. Beattie lectures internationally and consults for large organizations in the area of voice and data communications, networks and troubleshooting.

Other Faculty: Mr. E. Braun, Mr. D. Freberg, Mr. J. Scott, Mr. P. Smith and Mr. L. Warner.

■ Digital Communication Systems p. 14

Dr. Daniel Bukofzer has worked extensively in the areas of digital and high-speed analog communication and transmission systems. His application experience has included the development of all-digital multiplexors and demultiplexors, digital code converter circuits, jamming simulators, techniques for data deconvolution, and forward error correction techniques.

Other Faculty: Dr. D. Bensoussan, Dr. M. Quinn and Dr. M. Roden.

*Each course is taught by the author or one of a team of highly skilled instructors.



Learn from Experts

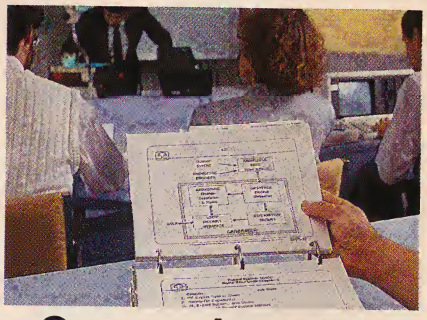
These courses are developed and taught by technical professionals . . . for technical professionals. You'll benefit from a well-organized presentation that emphasizes the practical application of state-of-the-art technology.



Effective Methods

Dual overhead projectors insure rapid communication. Where appropriate, slides, films, videotapes and live demonstrations augment the presentation.

In many courses, computer-based training with hands-on exercises, lab sessions and small team workshops provide concrete "Learn-by-Doing" experience.



Comprehensive Materials

Comprehensive Course Notes replicate each projected transparency, permitting full concentration on what is being taught.

The "State-of-the-Art Supplemental Readings" amplify key course topics, and provide a valuable "after course" reference.

PRODUCTIVITY THROUGH EDUCATION®

JB

13 COURSES ON

DATACOMM & NETWORKS PLUS ARTIFICIAL INTELLIGENCE

HOW TO ENROLL

PHONE:

TOLL FREE: (800) 421-8166

INTEGRATED COMPUTER SYSTEMS

East Coast Office
(703) 847-6448

West Coast Office
(213) 417-8888

WRITE:

8000 Towers Crescent Dr.
Suite 300
Vienna, VA 22180
Fax: (703) 847-6319

5800 Hannum Ave.
P.O. Box 3614
Culver City, CA 90231-3614
Telex: 4995168 ICS LA
Fax: (213) 410-2952

Phone Today to Reserve Your Place:

Class sizes are limited and classes are often filled to capacity. We recommend that you enroll early.

Immediate Confirmation:

You will receive immediate confirmation of enrollment and course location details by phone (or telex) and subsequently by mail. You can reserve your place without obligation. There is no cancellation fee if you notify us at least two weeks before the course. If a course is cancelled for any reason, you will be notified at least two weeks in advance.

$100 Prepayment Discount:

While our long-standing policy is that payment is not required until after you attend the course, we are pleased to offer a $100 discount for prepayment. Payment may be made by check or major credit card: VISA MasterCard AMERICAN EXPRESS

$100 Multiple Enrollment Discount:

A $100 discount for each participant is provided when 2 or more from the same organization enroll and are invoiced together. Each gets the discount even if he or she enrolls for different courses or different sessions of the same course. PLEASE NOTE: This discount is available *in addition* to the $100 prepayment discount.

Please take advantage of these discounts and call today to enroll.

Course Fee:

The course fee includes a comprehensive set of course materials, and morning and afternoon refreshment breaks each day. These course materials include a detailed five-level set of course notes that parallels the presentation, case studies, all workshop session exercises, and a set of State-of-the-Art Reprints.

Course	Title	Length	Discount Prepaid Fee	Standard Fee
253:	PC Networking: Hands-On Workshop HANDS-ON	3 Days	$995	$1095
256:	Micro-Mainframe Connectivity DEMO LABS	3 Days	$995	$1095
350:	Introduction to Datacomm & Networks	4 Days	$1295	$1395
352:	Local Area Networks: Implementation & Configuration	4 Days	$1295	$1395
353:	Computer Network Management & Control	4 Days	$1295	$1395
354:	Integrated Voice/Data Communications & ISDN	4 Days	$1195	$1295
355:	Computer Network Design: Protocols & Layered Architectures	4 Days	$1195	$1295
356:	Troubleshooting Datacomm & Networks HANDS-ON	4 Days	$1295	$1395
357:	MAP/TOP: Technical Introduction & Implementation Status Report	4 Days	$1195	$1295
358:	Digital Communication Systems: Design & Applications	4 Days	$1195	$1295
390:	Knowledge-Based Systems & Artificial Intelligence HANDS-ON	4 Days	$1195	$1295
392:	Expert Systems Design Workshop HANDS-ON	4 Days	$1195	$1295
440:	Fiber Optic Communication Systems: Design Tools & Techniques	4 Days	$1295	$1395

Course Hours & Completion Certificate:

Orientation (first day): 8:30am–9:00am
Class Hours: 9:00am–4:30pm (last day 9:00am–3:30pm)
Informal Discussion with Instructor: 4:30pm–5:30pm

Each attendee receives a diploma awarding a Continuing Education Unit (CEU) for each 10 hours of class participation.

BRING THESE COURSES TO YOUR FACILITY



▲ ***WHEN PRESENTED AT YOUR OWN FACILITY, THESE COURSES*** *provide even greater cost savings. Tuition is reduced, and travel expenses are eliminated, making it cost-effective to train entire teams.*

You'll Benefit From . . .

Proven, successful courses with expert instructors and comprehensive materials. For Hands-On and Demonstration Courses, we bring all of the in-class computers and electronic equipment to your facility.

A Major Advantage

Courses at your facility can be customized. Specific topics can be emphasized according to the needs of your organization, and timing can be adjusted to fit your schedule. By training an entire team, you'll accelerate the beneficial impact of new technology within your organization.

Experience Produces Results

Over the last 12 years, we have presented thousands of On-Site Courses for hundreds of organizations including: IBM, AT&T, GE, the US Navy, Tektronix, Hughes, HP, and TI. These and other references are available.

Take the First Step Today!

Courses at your facility are surprisingly easy and quick to organize. The first step can be taken by the technical staff, training department, or management.

For Courses at Your Facility Phone:

Anne-Lyse Heron or Reta Singer
On-Site Coordinators
(800) 421-8166 or
(213) 417-8888

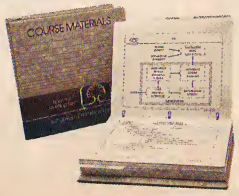
We'll provide you with all the assistance and information you need to bring these high-impact courses to your organization.

Call today for a complete set of three brochures with details on 39 Technical Short Courses: (800) 421-8166.

Priority Code:

To facilitate enrollment, please give our registrar the Priority Code:

PCODE 5735



To Order Course Materials Separately:

Each participant receives a comprehensive set of course materials included in the course fee. If you are unable to attend, you may purchase the course materials separately.

Price: $275 per course (includes regular UPS shipping within USA)

Sales Tax: California residents please add 6% (6½% in Los Angeles County)

© Copyright Integrated Computer Systems 1988.
8805UB

AIRFARE DISCOUNTS

NEW

AmericanAirlines

in cooperation with



INTEGRATED COMPUTER SYSTEMS

offers an exclusive

5% off ANY Air Fare or

up to 45% off Full Coach Fares when travelling to these courses.

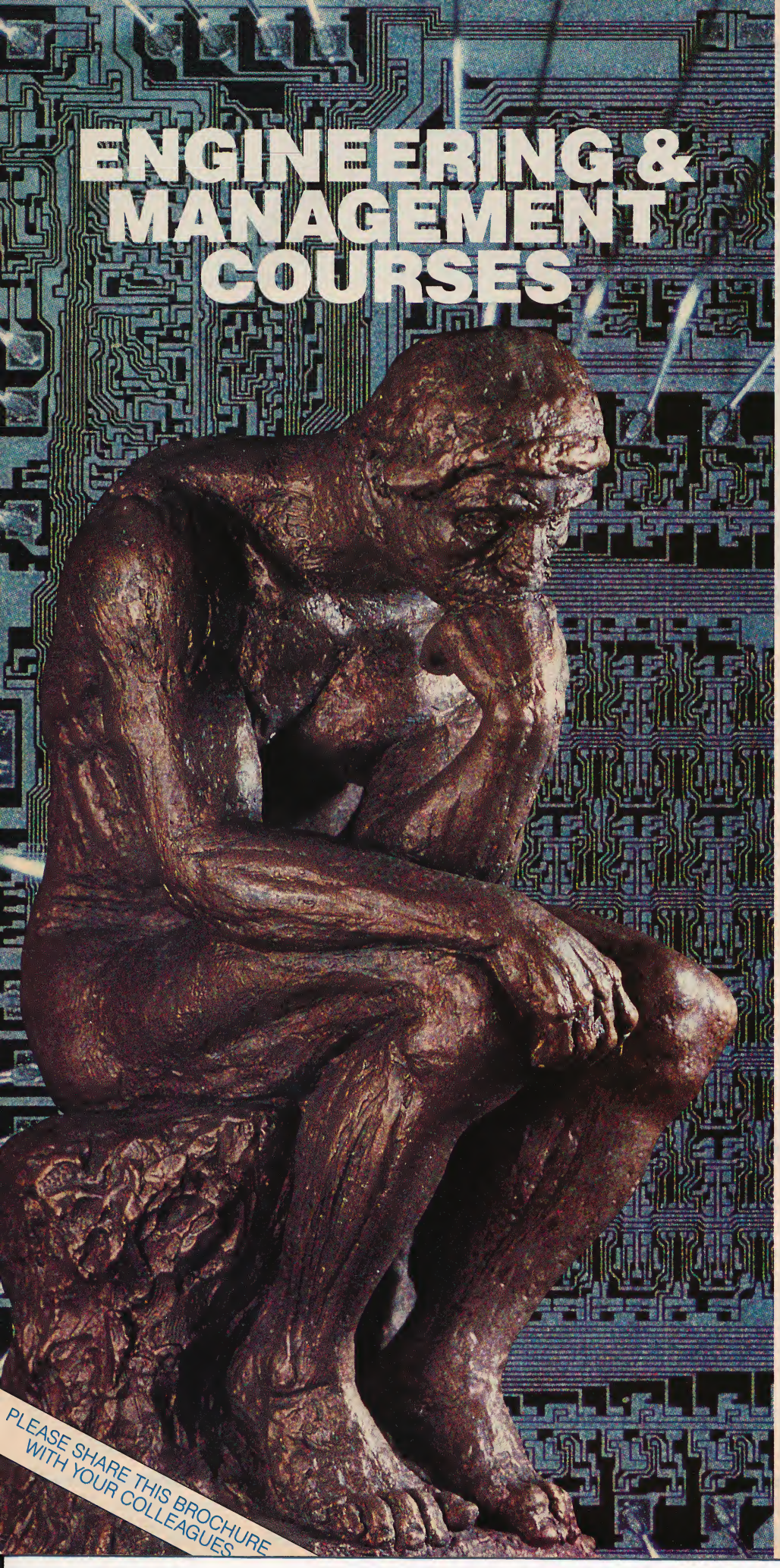
To take advantage of these exclusive low fares, call us today and enroll!

TOLL FREE 800-421-8166

Discounts subject to availability. Some restrictions may apply.

SAVE

SAVE



ENGINEERING & MANAGEMENT COURSES

PLEASE SHARE THIS BROCHURE WITH YOUR COLLEAGUES

SHORT COURSES

May 1988-December 1988

Microprocessors

- Microprocessor Software, Hardware & Interfacing: Hands-On Introduction HANDS-ON 2
- Microprocessor Troubleshooting: Hands-On Tools & Techniques HANDS-ON 3
- 16 & 32-Bit Microprocessor System Design 4

Digital Processing & Applications

- Digital Signal Processing: Techniques & Applications DEMO LABS 5
- Image Processing & Machine Vision DEMO LABS 6
- Digital Signal Processing: Real-Time Hardware Design 7
- Digital Control Systems: Design Techniques & Tools 8

Computer Architectures

- Supercomputers, Hypercubes & High Performance Architectures 9

Management Skills

- Effective Skills for Technical Managers 10
- Software Project Management Tools & Techniques DEMO LABS 11

NEW! Project Management: Planning, Scheduling, Tracking & Controlling HANDS-ON 12

Technical Planning

NEW! System Engineering for Integrated Hardware/Software Applications 13

- Security for Computer and Communication Systems 14

Enrollment & On-Site Courses 16

from the learning tree®

INTEGRATED COMPUTER SYSTEMS

EDUCATION IS OUR BUSINESS®

8805UB

Microprocessor Software, Hardware & Interfacing: Hands-On Introduction

HANDS-ON

Course 161

8805 UB 161



4 Days

TORONTO
May 31-June 3
LOS ANGELES
June 7-10
WASHINGTON, D.C.
July 12-15
SAN DIEGO
August 9-12
BOSTON
August 23-26
WASHINGTON, D.C.
September 13-16
SAN FRANCISCO
September 20-23
LOS ANGELES
October 18-21

Courses at Your Facility
Please Inquire

Course Benefits

This unique **hands-on course** provides a broad foundation in the skills required for the design, programming, and real-world interfacing of microprocessor applications. Techniques developed in the lectures are immediately implemented in class by each participant on the **Microcomputer Training System provided for their sole use throughout the course.**

The combination of detailed classroom presentations and the execution of practical hands-on experiments provides an intensive learning experience which insures that fundamental techniques and principles will be fully understood.

YOU WILL LEARN HOW TO:

- **Configure Microcomputer Hardware at the Chip Level**
- **Program Microprocessors at the Machine Level**
- **Interface to Sensors, Actuators, and Other External Devices**
- **Use Interrupts, Interval Timers, and A/D Converters**
- **Make Hardware/Software Applications Tradeoffs**
- **Compare Assembly and High-Order Language Implementations**
- **Apply Basic Software Engineering Tools and Techniques**

WHO SHOULD ATTEND:

Project and design engineers, technicians, programmers, systems analysts, and technical managers who will be involved in implementing microprocessor applications. **No prior background in microprocessor or programming technology is required;** however, an engineering/technical orientation is assumed.

Course Content

1. MICROPROCESSOR SYSTEM OVERVIEW

Basic Concepts & Terminology

Microprocessor System Components
- The Microprocessor ■ Memory
- I/O & Interfacing ■ Software

2. HANDS-ON PROGRAMMING

The Microprocessor's Instruction Set
- Data Movement & I/O Operations
- Jumps & Loops ■ Arithmetic Operations
- Conditional Instructions ■ Subroutines

Implementing Commonly Used Programs
- Timing Routines ■ Controller Functions
- Arithmetic ■ I/O Buffer Software

Using a Two-Pass Assembler

3. MICROPROCESSOR SYSTEM HARDWARE

Microprocessor, Clock & Control Logic
- Power-On Circuits ■ Clock Logic

System Bus Design Considerations
- Data, Address & Control Lines
- Bus Transceivers & Drivers
- Interpreting Timing Diagrams

Designing the Memory Section
- Read-Only Memory: ROM, EPROM, E^2PROM
- Static vs. Dynamic RAM
- Refresh Control
- Address Decoding Schemes
- Battery Back-Up Techniques

4. HANDS-ON INTERFACING WORKSHOP

Using Parallel I/O Ports
- Front Panel Control ■ Reading Switches
- Interfacing to Displays ■ Motor Control

Using Serial I/O Ports
- Serial Interface Standards — RS232
- ASCII and EBCDIC Codes
- Designing a UART-Terminal Interface

Interfacing with Analog Devices
- Applications of D/A and A/D Converters
- External Device Isolation
- Opto-Couplers ■ Voltage Level Conversion

Using Interval Timers
- Real-Time Programming
- Accurate Interval Timing
- Watchdog Timers

Interrupt & DMA Techniques
- Vectoring ■ High-Speed Data Transfer
- Priority Interrupt Controller Chips

Using Advanced Peripherals

5. ADVANCED MICROCOMPUTER CONCEPTS

8- & 16-Bit Microcomputer Architectures
- Address Space ■ Memory Management
- Bus Systems ■ Co-Processors
- Advanced Peripheral Chips

Software Engineering Methods
- Structured Programming ■ Modular Design
- Pseudo-Code ■ Flow Chart Alternatives

Microcomputer Development Tools
- Single- & Multi-User Development Systems
- Using Cross-Compilers to Develop Software on Minis & Mainframes
- Text Editors ■ Assemblers & Compilers
- Linkers & Loaders ■ Debuggers
- In-Circuit Emulation for Software & Hardware Integration & Debugging

6. DEVELOPING AN APPLICATION

Analyzing System Requirements
- Processor Throughput
- Memory Size & Organization
- Interface & I/O Specifications

Hardware/Software Design Tradeoffs

Selecting the Right Microprocessor
- Key Factors ■ Vendor Analysis
- 8-Bit vs. 16-Bit vs. Single-Chip Micros
- Software Support ■ Development Tools

Choosing the Right Language
- Assembly vs. High-Order Languages
- Programming Time vs. Memory vs. Speed
- PASCAL ■ FORTRAN ■ PL/M ■ C

7. FUTURE HARDWARE & SOFTWARE TRENDS

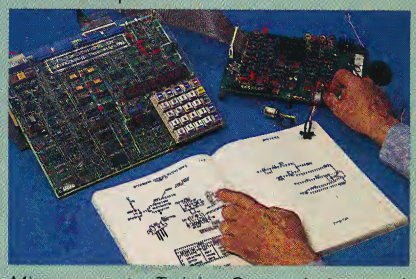
HANDS-ON

Each participant receives a Microcomputer Training System for use throughout the course. The Training System includes a keyboard and display for program entry and execution, plus a wide range of interfaces including: timers, D/A and A/D converters, optical couplers, power transistors, parallel and serial ports, and a multi-priority interrupt system.

In-class experiments include:

- **Controlling an 8-Digit Display**
- **Debugging by Single Step and Breakpoints**
- **Implementing Timing Functions in Software & Hardware**
- **Using I/O Ports**
- **Controlling a Robot Arm**
- **Interfacing with Analog Inputs and Outputs**
- **Single Interrupts/Waveform Generation**
- **Multiple Device Interrupts**

In addition to the in-class exercises, interactive workshops give participants the opportunity to develop their own projects. Homework assignments further reinforce each concept.



Microcomputer Training System being used for in-class programming and interfacing exercises.

INTEGRATED COMPUTER SYSTEMS

ENROLL NOW!

Call Toll Free **(800) 421-8166** or

East Coast Office **(703) 847-6448**

West Coast Office **(213) 417-8888**

8805 UB 142

Microprocessor Troubleshooting: Hands-On Tools & Techniques

Course 142 HANDS-ON



4 Days

WASHINGTON, D.C.
May 24-27
TORONTO
June 7-10
LOS ANGELES
June 21-24
WASHINGTON, D.C.
July 19-22
SAN DIEGO
August 16-19
BOSTON
August 30-September 2
WASHINGTON, D.C.
September 20-23
LOS ANGELES
October 25-28
SAN FRANCISCO
December 13-16

Courses at Your Facility
Please Inquire

Course Benefits

Effective troubleshooting of microprocessor systems requires specialized skills and equipment. This unique course teaches **practical** troubleshooting techniques reinforced by **in-class, hands-on training** with test equipment specifically intended for microprocessor applications.

YOU WILL LEARN HOW TO:

- **Write Simple Programs for Use in Troubleshooting**
- **Troubleshoot Processor, Bus, Memory and I/O Failures**
- **Isolate Hardware vs. Software Faults**
- **Use Signature Analyzers, In-Circuit Emulators, Logic Analyzers, and Other Troubleshooting Equipment**
- **Reduce Troubleshooting Time and Cost**
- **Evaluate Commercial Troubleshooting Equipment**

Course participants will return to their jobs prepared to implement effective troubleshooting techniques for their own applications. The course is field-proven; over 4,000 former participants have reported that the course successfully met the above objectives.

WHO SHOULD ATTEND:

This course is valuable for technicians, engineers, and managers involved in the design, production testing, or field service of microprocessor-based systems. While no prior hands-on experience with microprocessors is required, a basic knowledge of digital logic and microprocessor architecture is assumed.

Course Content

1. MICROPROCESSOR FUNDAMENTALS

Basic Concepts and Terminology
- The Microprocessor Structure
- Memory Design & Address Decoding
- The System Bus: Data, Address, Control
- I/O Fundamentals ■ Interrupt Systems
- Overview of Popular Microprocessors

Hands-On Introduction to Programming
- Register & Memory Operations
- Basic I/O ■ Jumps & Loops
- Subroutines ■ Stack Operation

2. OVERVIEW OF TROUBLESHOOTING

What's Different About Microprocessor Systems?
- Software-Controlled Functions
- Bus Structures & Device Complexity
- Static and Dynamic Logic
- Tri-State Signals and Components

Fault Identification
- Microprocessor, Clock, RAM, ROM, I/O
- Shorts & Opens ■ Marginal Design
- Bus/Tri-State Problems ■ Noise
- Intermittents ■ Handshaking Problems

Troubleshooting Methodology
- Why Traditional Methods Don't Work
- Hardware & Software Approaches
- Microprocessor-Oriented Techniques

3. BASIC TOOLS & TEST EQUIPMENT

Fundamental Testing Techniques
- Logic Analyzers ■ Signature Analyzers
- In-Circuit Emulators ■ Logic Probes
- Production Test & Diagnostic Systems
- Diagnostic Software

Major Considerations & Tradeoffs
- Cost ■ Capability ■ Skill Level
- Set-Up Time ■ Ruggedness ■ Portability

Using Oscilloscopes
- Controlling the System–Scope Loops
- Observing Multichannel Activity
- Trigger Sources ■ Limitations

4. USING DIAGNOSTIC SOFTWARE

Basic Principles
- Diagnostic vs. Functional Testing
- Built-in Diagnostics ■ Plug-in PROMS
- Diagnostics with In-Circuit Emulation
- High-Level Language Diagnostics

Specific Diagnostic Problems
- Fault Coverage
- Microprocessor CPU Test Programs
- RAM, ROM, I/O Device Tests

5. USING LOGIC ANALYZERS

How to Use Logic Analyzers for:
- Timing Analysis ■ Tracing Program Flow
- Software Problems ■ Memory Failures
- I/O Failures ■ Intermittents ■ Glitches

Survey/Selection Methodology
- Triggering Capability ■ Display Format
- Speed, Storage, Channel Capacity
- Synchronous vs. Asynchronous Modes
- Available Probe Types and Options
- Component Signal Level Capabilities

6. USING IN-CIRCUIT EMULATORS

How In-Circuit Emulation Works
- Case Study: The Fluke 9010A

Using In-Class Fluke Emulators for:
- Debugging a "Totally Dead" System
- Checking Bus Integrity
- Identifying Chip-Level Faults
- Developing Test Software
- Running Protected Diagnostics

7. APPLYING SIGNATURE ANALYSIS

Debugging with Signature Analysis
- Operation ■ Capabilities
- Limitations ■ Chip-Level Isolation
- Signature vs. Logic Analyzers
- Programming & Hardware Needs
- Documentation Methods

8. PROBES & CURRENT TRACERS

- Logic Probes ■ Logic Pulsers
- Current Tracers ■ Locating Shorts

9. OVERVIEW OF AUTOMATIC TEST EQUIPMENT (ATE)

- Cost vs. Yield Factors
- Shorts & Opens Testers
- Functional vs. In-Circuit Testers
- Portable Field Service ATE

10. COURSE REVIEW & SUMMARY

HANDS-ON

On the first day each participant is provided with a **Microcomputer Training System** to perform hands-on exercises that reinforce basic microprocessor hardware and programming concepts.

Participants are thereafter grouped in small teams. Each team is provided with a complete **troubleshooting station** which includes: signature analyzer, in-circuit emulator with diagnostic software, and microcomputer with selectable hidden faults.

Hands-on exercises include identifying:

- **CPU, Clock & Control Logic Failures**
- **Data & Address Bus Shorts & Opens**
- **PROM & RAM Faults**
- **Chip-Select Failures**
- **Interrupt Operation ■ I/O Circuit Faults**
- **Keyboard Display Errors**
- **SW Bugs Disguised as HW Faults**

Through hands-on exercises under the instructor's guidance, students learn effective techniques for using each type of equipment to diagnose the failures most often encountered in actual practice.



Participants use signature analysis, diagnostic software and in-circuit emulation during the in-class troubleshooting exercises.

INTEGRATED COMPUTER SYSTEMS

ENROLL NOW!

Call Toll Free **(800) 421-8166** or

East Coast Office **(703) 847-6448**

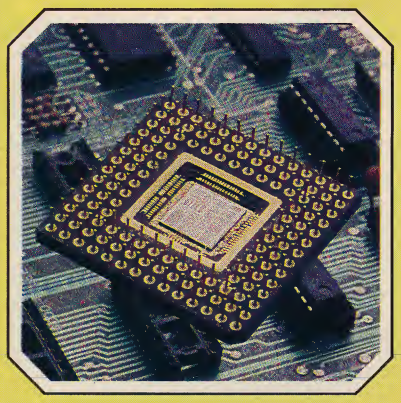
West Coast Office **(213) 417-8888**

80386 in photo courtesy of Intel.

8805 UB 166

16 & 32-Bit Microprocessor System Design

Course 166



4 Days

TORONTO
May 31-June 3
WASHINGTON, D.C.
June 7-10
SAN DIEGO
July 19-22
OTTAWA
August 9-12
WASHINGTON, D.C.
August 16-19
LOS ANGELES
September 13-16
BOSTON
October 18-21
SAN DIEGO
November 1-4

Courses at Your Facility
Please Inquire

Course Benefits

The advanced features of today's 16 and 32-bit microprocessors allow them to achieve speeds 10 to 100 times that of 8-bit micros. When appropriately applied, facilities such as cache memory, pipelining, co-processors and context switching can significantly improve the performance of micro-processors in real-time applications.

This course provides participants with a detailed understanding of how to use these powerful features to their best advantage, and how to get the most out of today's advanced development software and systems. The course emphasizes alternative approaches for solving advanced system design problems, and provides concrete guidelines for implementing 16 and 32-bit designs.

YOU WILL LEARN HOW TO:

- **Take Full Advantage of the Features of 16 and 32-Bit Microprocessors**
- **Select the Most Appropriate 16 or 32-Bit Microprocessor**
- **Use Co-Processors and Intelligent Peripheral Interfaces to Significantly Improve System Performance**
- **Utilize Memory Management Chips to Implement Task Segmentation, Virtual Memory, and Improved Addressing Schemes**
- **Evaluate Development Tools, Languages, and Operating Systems**

WHO SHOULD ATTEND:

This course is valuable for hardware and software engineers, systems designers, programmers and technical managers who are involved in developing micro-processor applications.

Course Content

1. INTRODUCTION & DESIGN OVERVIEW

Basic Concepts and Terminology
- Advanced Microprocessor Architectures
- Addressing Methods and Bus Structures
- Instruction Sets & Programming Techniques

Introduction to Advanced Microprocessors
- Survey of Suitable Applications
- Partitioning: CPU vs. Support Chips
- Software Capabilities of Advanced Chips

Key Features of 16/32-Bit Micros
- Segmentation ■ Memory Management
- Coprocessors and Multiprocessors
- Throughput and Processing Speeds
- Fast Execution of Higher Order Languages
- Microprocessor Families: Intel 80xx(x), Motorola 680x0, National 320xx, Zilog 800x(x), TMS 320 series

Applications Overview
- Avionics ■ Process Control ■ DSP
- Embedded Systems ■ Proprietary Products
- CAD/CAM Workstations ■ Graphics

2. DESIGNING WITH 16 & 32-BIT MICROS

Making Use of Advanced CPU Capabilities
- Memory- vs. Register-Based Structures
- Internal/External Arithmetic Capabilities
- Boosting Speed With Prefetch & Pipelining
- Determining When Cache Memory is Useful

Using Memory Management Facilities
- Logical/Physical Address Mapping Methods
- HW/SW Implementation of Virtual Memory
- Access Protection ■ Recovery from Faults

Technology Update
- Implementing Modules & Stack Frames
- In-Depth Analysis of the Intel Family
- Features & Evolution of the 680x0 Family

3. BUS DESIGN TRADEOFFS

Fundamental Concepts
- The Rationale for Buses
- Basic Bus Capabilities
- Software Support

Surveying the Major Buses
- Current Popularity, Usage & Future
- General & Special Functions of Available Cards

Issues in Bus Implementation
- Master/Slave Modules ■ Interrupts
- Location Monitor ■ Message Passing
- Mailboxes ■ Bus Expanders
- Bus Interfacing: the Real Estate Problem
- IBM PC/AT BUS ■ IBM Microchannel

VME Buses
- Fundamental Design Parameters
- Families of VME Boards

MULTIBUS I & II
- Basic Concepts & Approaches
- Intel's Message Passing
- Geographic Addressing Techniques

4. DESIGNING HIGH-SPEED SYSTEMS

Coprocessors and Special Purpose Chips
- Using Math and Floating-Point Chips
- Managing Paged Memories
- Intelligent I/O and Slave Processors
- Designing External Cache Memory

Real-Time Considerations
- Defining Real-Time
- Designing the Interrupt System: Context Switching, Asynchronous Interrupts, Synchronous Interrupts & SW Trapping
- Choosing and Using a Kernel

Developing Multiprocessor Designs
- Tightly vs. Loosely Coupled Designs
- Interprocessor Communications Software, Transparent Communication Strategies
- Using Multiport RAM & Shared Memory
- Multiprocessor Arbitration Circuits

Reduced Instruction Set (RISC) Designs
- Complex vs. Reduced Instruction Sets
- IBM/RT & Hewlett-Packard Spectrum
- The Transputer

5. INTERFACING & COMMUNICATIONS

Interfacing with Common Peripherals
- Serial & Parallel I/O Designs
- Writing & Installing Drivers
- Hardware/Software Techniques for DMA
- Disk Controllers & Graphics Interfaces
- Small Computer Systems Interface - SCSI
- Other Peripheral Buses

Computer-to-Computer Communications
- Layered Communications Software
- Using Ethernet

6. ADVANCED DEVELOPMENT TOOLS & HOLs

Evaluating Today's Development Tools
- Configurations: PC-Based, Minis, Workstations & Self-Hosted Systems
- Software Development Tools
- Software Trace & Simulation Capabilities
- Advanced μP In-Circuit Emulators

Operating Systems for SW Development
- UNIX ■ MS-DOS ■ VAX/VMS
- Intel RMX-x86 ■ OS/2

Comparison of High-Order Languages
- Language/Application Compatibility
- Compilation Time ■ Execution Speed
- Support of Structured Programming
- Ada ■ C ■ FORTH ■ FORTRAN ■ PASCAL
- Structured Assembly Language
- Assembly/HOL Interfacing

7. SELECTION CRITERIA & TRADEOFFS

Selecting a Family
- Architectural Comparisons
- Capabilities of Support Chips
- Board-Level Module Availability
- Development System Capabilities
- Languages and Support Software
- Second-Source & Third-Party Support

Programming Benchmarks
- How to Get Meaningful Benchmarks
- Interpreting Manufacturer's Benchmarks: Carnegie-Mellon & Drystone

Design Case Studies

System Design Problems are used throughout the course to illustrate the choices a Designer must make.

Case Studies include:

- **Design Considerations for Harsh Environments**
- **Card Selection Criteria for Remote Monitoring**
- **Packaging Constraints**
- **Simulation: Choosing the "Best" Language and Development System**

INTEGRATED COMPUTER SYSTEMS

ENROLL NOW!

Call Toll Free **(800) 421-8166** or

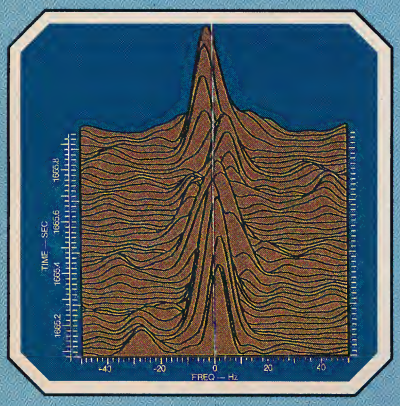
East Coast Office **(703) 847-6448**

West Coast Office **(213) 417-8888**

Digital Signal Processing: Techniques & Applications

DEMO LABS

Course 412



4 Days

BOSTON
June 14-17

SAN DIEGO
June 21-24

WASHINGTON, D.C.
June 28-July 1

WASHINGTON, D.C.
August 30-September 2

SAN DIEGO
September 13-16

BOSTON
October 4-7

TORONTO
October 4-7

LOS ANGELES
October 18-21

WASHINGTON, D.C.
November 1-4

Courses at Your Facility
Please Inquire

Course Benefits

8805 UB 412

Digital signal processing (DSP) techniques have been revolutionized by the advent of powerful, programmable VLSI devices and inexpensive digital computers/array processors. Sophisticated real-time signal processing systems delivering superior performance, flexibility and reliability at significantly reduced cost are currently being implemented.

This course integrates the fundamental algorithms of digital filters and spectral analysis with advanced software techniques and hardware architectures.

YOU WILL LEARN HOW TO:

- **Apply Digital Signal Processing Fundamentals**
- **Use Fast Fourier Transform (FFT) Algorithms**
- **Design High-Speed FIR and IIR Filters**
- **Implement Filters in Hardware and Software**
- **Utilize Spectral Estimation Techniques**
- **Apply Special-Purpose VLSI and Array/Pipeline Processors**

WHO SHOULD ATTEND:

The course is valuable for project and design engineers, programmers and technical managers who are responsible for the implementation of advanced digital signal processing systems, and for those whose responsibilities require a understanding of both the advanced techniques available today and their potential for use in a wide variety of applications.

COMPUTER PROGRAMS

Participants receive FORTRAN Source Listings as part of the course material. These programs permit the user to specify design parameters and produce **complete, computer-designed digital filters.**

Course Content

1. INTRODUCTION

Basic Concepts & Terminology
- What Is Digital Signal Processing (DSP)?

Digital vs Analog Signal Processing
- Performance ■ Cost ■ Speed
- Signal Drift ■ Guaranteed Accuracy

Overview of Available Hardware
- Minicomputers
- Single-Chip Digital Signal Processors
- Array Processors ■ Microprocessors
- Microprogrammable Devices
- A/D Converters, Multipliers
- TMS 320 Family & Other DSP Chips

Applications Overview
- Image and Speech Processing
- Telecommunicatoins
- Radar & Sonar Processing
- Test Equipment
- Industrial Control ■ Biomedical Analysis

2. BASIC TOOLS & TECHNIQUES FOR DSP

The Signal
- Describing the Signal
- Time & Frequency Domains
- Sampling, Bandwidth & Aliasing
- Sampling Bandpass Signals

The System
- What Is a System?
- Convolution & the Linear System
- Recursive/Nonrecursive Difference Equations
- Frequency Response ■ The Z-Transform
- Discrete Fourier Transform
- Practical Examples and Uses

Fast Fourier Transform (FFT) Algorithms
- Use in Spectral Analysis, Detection
- The FFT Algorithm & Variations
- The New Winograd Fourier Transform (WFT)

3. FINITE IMPLUSE RESPONSE (FIR)—TRANSVERSAL FILTERS

Characteristics of FIR Filters
- Stability ■ No Phase Distortion
- Powerful, Simple Design

When to Use FIR Filters

How to Design FIR Filters
- Remez Exchange Algorithm (Optimal)

Implementing FIR Filters
- Tapped Delay Lines, CCDs
- Direct and Indirect Convolution
- Stack Data Structures in RAM & ROM
- How to Select the Method

Practical Examples
- Multi-Pass/Stop Band Filters

4. INFINITE IMPLUSE RESPONSE (IIR)—RECURSIVE FILTERS

Characteristics
- More Efficient than FIR
- Phase Distortion and Group Delay Effects

When to Use IIR Filters

Design Techniques
- Superiority of the Bilinear Method

Implementation
- Direct Recursion
- Cascaded Second Order Sections
- Programming Considerations

Simulation of Analog Systems

Examples
- Butterworth ■ Chebyshev ■ Elliptic

5. APPLYING STATISTICAL TECHNIQUES

General Considerations
- Random Sequences
- Autocorrelation Functions
- Power Spectral Density

Optimal Mean Square Filters
- Wiener Nonrecursive
- Adaptive LMS Algorithms
- Introduction to Kalman Filters

Detecting Signals in Noise
- Designing the Matched Filter
- Flase Alarm/Detection Probabilities

6. SPECTRAL ESTIMATION TECHNIQUES

Applications
- Radar/Sonar ■ Vibration Analysis
- Speech Compression ■ Seismic Analysis

Applying Fourier Transform Methods
- Windowing and Transforming
- Sidelobes: Problems and Resolution
- Averaging and Variance Considerations

Maximum Entropy Method (MEM)
- Maximum Entropy Spectral Analysis
- Super Resolution
- Speech Compression (LPC)

7. SPECIFYING THE HARDWARE

Method Selected vs. Hardware Required
- Multiplier Speed ■ Memory Needs
- Matching the Hardware with the Algorithm

Effects of Digital Word Length
- ADC Noise ■ Coefficient Quantization
- Multiplier Roundoff Effects

Hardware Survey
- ADC, DAC ■ Multipliers/Adders
- Capabilities of Single-Chip DSP Devices vs. Microprogrammed Processors
- Array Processors ■ Spectrum Analyzers
- Single-Board Real-Time Processing Units

8. FUTURE TRENDS

- Expected Hardware Developments
- Developments in Processing Techniques

DEMO LABS

Throughout the course, key concepts and algorithms are implemented on an in-class computer.

Participants will see demonstrations of software and applications that:

- **Apply Fast Fourier Transforms**
- **Perform Spectrum Analysis**
- **Design IIR and FIR Filters**



INTEGRATED COMPUTER SYSTEMS

ENROLL NOW!

Call Toll Free **(800) 421-8166** or

East Coast Office **(703) 847-6448**

West Coast Office **(213) 417-8888**

Image Processing & Machine Vision

DEMO LABS

8805 UB 411

Course 411



4 Days

TORONTO
June 14-17
WASHINGTON, D.C.
June 21-24
SAN DIEGO
June 28-July 1
LOS ANGELES
July 26-29
TORONTO
July 26-29
BOSTON
August 30-September 2
WASHINGTON, D.C.
September 20-23
SAN FRANCISCO
October 4-7
LOS ANGELES
November 1-4

Courses at Your Facility
Please Inquire

Course Benefits

Sophisticated Image Processing and Machine Vision techniques are now available for an increasing array of industrial, military and medical applications such as manufacturing, autonomous guidance systems, and diagnostic procedures. The emergence of low cost solid state sensors and very high speed integrated processing chips, coupled with powerful new algorithms and software, has led to tremendous growth in the imaging and vision industries.

This course provides a thorough understanding of the fundamentals of image processing and machine vision, including key techniques for their implementation and integration in practical, cost-effective, real-world applications. Throughout the course, participants will observe actual imaging and recognition procedures demonstrated on the In-Class Image Processing Workstation.

YOU WILL LEARN HOW TO:

- **Apply the Tools and Techniques of Image Processing**
- **Determine and Specify Machine Vision System Requirements**
- **Develop Image Processing and Recognition Algorithms**
- **Evaluate Hardware/Software and Performance Trade-offs**
- **Successfully Develop Imaging and Machine Vision Applications**

WHO SHOULD ATTEND:

This course is intended for engineers, scientists, programmers, systems analysts, and project managers who are involved in the design, development or application of image processing and machine vision systems.

Course Content

1. INTRODUCTION AND OVERVIEW

Fundamental Concepts
- Basic Definitions and Terminology
- Image Sensing, Processing and Analysis
- 2-D and 3-D Machine Vision Tasks
- Design and Implementation Methodology
- State-of-the-Art Tools and Techniques

Image Processing and Vision Systems
- Solid State and Laser Vision Sensors
- Special VLSI Components and Architectures
- Low Cost Workstations & PC Based Systems
- Networked Image Information Systems

Applications Overview
- Imaging Systems in Business and Industry
- Vision Assisted Vehicle Guidance
- Automatic Target Recognition and Tracking
- Imaging Diagnostics in Medicine & Biology

2. FUNDAMENTAL ELEMENTS AND PROCEDURES

Image Acquisition Tools and Techniques
- Image Sensor Characteristics & Performance
- Color and Multispectral Imaging
- Structured Light and Stereo Ranging
- Detecting and Characterizing Motion

The Human Visual System
- Physiology and Psychology of Human Vision
- Perception of Contrast, Sharpness, Color
- Characterizing Image Quality

Processing, Transmission and Display
- Image Enhancement and Restoration
- Compression for Storage and Transmission
- Softcopy and Hardcopy Displays, Halftoning
- Matching the Human Visual System

Image Analysis and Recognition Techniques
- 2D vs. 3D Machine Vision
- Separating Objects from Background
- Measuring Shape Features for Recognition
- Linear/Non-linear Discriminant Functions
- Knowledge-Based Recognition Techniques

3. IMAGE PROCESSING TECHNIQUES

Image Correction and Restoration
- Geometric Unwarping and Calibration
- Removing Aperture and Motion Blur
- Linear/Non-Linear Contrast Enhancement
- Filtering Details and Noise

Color Image Processing
- Additive vs. Subtractive Color
- Correcting and Enhancing Natural Colors
- False Color and Pseudocolor Techniques
- Color Capture, Display and Hardcopy

Image Coding and Compression
- Lossless vs. Lossy Image Compression
- Predictive and Transform Coding Techniques
- Document Coding, CCITT Group II and III
- Second Generation Coding Techniques

4. 2D AND 3D MACHINE VISION

Feature Extraction Techniques
- Levels of Symbolic Image Descriptors
- 2-D and 3-D Feature Measurement
- Edge, Line and Region Descriptors
- Advanced Features: Fourier and Moments
- Stereo and Structured Light Ranging

Designing Decision Algorithms
- Linear/Non-Linear Discriminant Functions
- Using Knowledge and Physical Constraints
- Recognizing Partially Hidden Objects
- Estimating Error Cost and Performance

3-D Scene and Motion Understanding
- Handling Complex or Cluttered Visual Tasks
- Hypothesis Generation and Testing
- Matching Sensed Images With 3-D Models
- Detecting and Characterizing Motion

5. IMPLEMENTING PRACTICAL SYSTEMS

System Engineering
- System Performance and Accuracy Goals
- Determining the Throughput Requirements
- Making a Computational Budget
- Identifying Concurrent Processes
- System Testing and Validation Procedures

Algorithms for Real-Time Systems
- Dealing with Real-Time Constraints
- Local Neighborhood Image Transformations
- Fast Fourier Transforms (FFTs)
- Correlation and Fast Template Matching
- Image Algebraic Operations

Special Purpose Architectures
- Cellular, Pipe-Line and Multi Processors
- Systolic Arrays, Hypercubes, Data Flow
- VLSI Chips, Components and Subsystems
- Trade-offs: Speed, Weight, Power, Cost

Software/Firmware Development
- Tools for Data and Algorithm Analysis
- Modular Functionality Specification
- Controlling Concurrent Processes
- Programming for System Flexibility

6. PRODUCT AND APPLICATION SURVEY

Available Systems and Components
- Sensors, Digitizers, Frame Grabbers
- Chip-Level Components and Subsystems
- Board-Level and Development Systems
- Turnkey Analysis and Vision Systems
- Image Workstations, PC Based Systems

Application Case Studies
- Picture Archival & Communications Systems
- Industrial Parts Measurement and Inspection
- Recognition of Fingerprints, Targets

DEMO LABS

Throughout the course, key concepts and algorithms are demonstrated on an IN-CLASS IMAGE PROCESSING WORKSTATION. Demonstrations include:

- **Digitizing Live Video Images**
- **Enhancing Contrast and Sharpness**
- **Performing Geometric Transformations**
- **Extracting Objects from Background**
- **Recognizing Specific Targets**



Image Processing and Recognition techniques are demonstrated on the In-Class Image Processing Workstation.

INTEGRATED COMPUTER SYSTEMS

ENROLL NOW!

Call Toll Free **(800) 421-8166** or

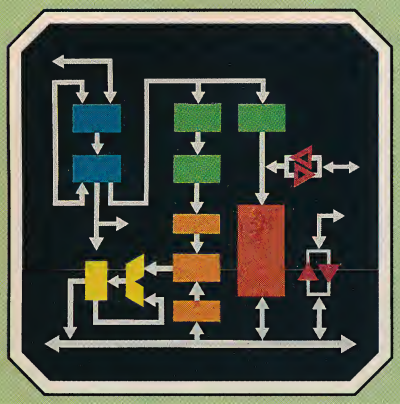
East Coast Office **(703) 847-6448**

West Coast Office **(213) 417-8888**

Digital Signal Processing: Real-Time Hardware Design

8805 UB 415

Course 415



4 Days

WASHINGTON, D.C.
May 24-27

LOS ANGELES
July 19-22

WASHINGTON, D.C.
August 9-12

OTTAWA
October 18-21

LOS ANGELES
December 13-16

WASHINGTON, D.C.
December 13-16

Courses at Your Facility
Please Inquire

Course Benefits

Recent advances in high-performance single-chip DSP processors and microprogrammable devices now offer engineers a wide range of solutions to the design of real-time signal processing (DSP) applications.

This course provides participants with a detailed understanding of how to implement a wide range of signal processing algorithms on Single-Chip DSP Processors and Microprogrammable Bit-Slice Devices, as well as 16 & 32-Bit Micros. The course emphasizes design tradeoffs and device selection.

YOU WILL LEARN HOW TO:

- **Partition Signal Processing Algorithms into Hardware, Software and Firmware Components**
- **Design the Hardware and Software to Implement Real-Time Digital Filters, FFTs, Correlation and Other Processes Using:**
 - **— Dedicated, Single-Chip Signal Processors**
 - **— Microprogrammable Bit-Slice Devices**
 - **— General Purpose 16 & 32-Bit Microprocessors**
- **Develop Programs and Microcode for Key Processors**
- **Evaluate Application Requirements and Select the Most Appropriate Hardware Technology and Components**

WHO SHOULD ATTEND:

This course is valuable for design and project engineers, software/firmware developers, systems analysts and technical managers who are involved with the design and implementation of digital signal and image processing systems. Familiarity with digital logic is assumed.

Course Content

1. INTRODUCTION & OVERVIEW

Fundamental Concepts
- Basic Definitions & Terminology
- Digital Signal Processing (DSP) Algorithms
- Programming vs. Microprogramming
- Parallelism & Pipelining Techniques

Basic Signal Processor Architectures
- Analog & Digital I/O ■ Bus Structures
- Data, Coefficient & Instruction Memory
- ALU's & Microprogrammed Sequencers

Hardware Implementation Alternatives
- Single-Chip Digital Signal Processors
- Microprogrammable Bit-Slice Devices
- 16 & 32-Bit Microprocessors

2. DIGITAL PROCESSING ALGORITHMS: AN ENGINEERING OVERVIEW

Implementation of DSP Algorithms
- The Fast Fourier Transform (FFT)
- Typical Filters: Recursive (IIR) & Non-Recursive (FIR)
- Convolution & Correlation in the Frequency and Time Domains

Computational Aspects of DSP Algorithms
- Number of Multiplies and Adds
- Data Structure & Address Generation
- Fixed vs. Floating-Point Computation
- Dynamic Range ■ Word Length
- Truncation & Round-Off Effects

3. DESIGNING WITH DEDICATED DSP CHIPS

Fundamental Concepts
- Complete "Digital-Processors-on-a-Chip"
- Architectures ■ Programming Methods
- Data Flow and Structures

Capabilities & Features of DSP Chips
- On vs. Off-Chip Programs & Data Memory
- Use of Multiple Data & Instruction Buses
- Independent Address Generators

Application Considerations
- Requirements for Support Chips
- Throughput ■ Preprogrammed Algorithms
- Programming Support Libraries, Assemblers, Compilers, Simulators

System Design with DSP Chips
- Architecture & Internal Resources
- Instruction Set & Program Model
- Circuit Design & Interfacing
- DSP Code and Benchmarks
- Algorithm Performance Analysis
- Integer & Floating Point DSP Chips:
 TI TMS 320C25, 32020, 32010
 Analog Devices ADSP-2100
 Motorola DSP5600
 ATT DSP-32
 TI TMS 320C30

Survey of DSP Chips
- Basic Architecture
- Typical System Configuration
- General Purpose Devices:
 ATT DSP-16
 Philips PCB 5010/11
 NEC μPD 77230, 7720
 OKI M6992
 Fujitsu MB 8764
- Algorithm-Specific DSP Chips:
 Zoran ZR34161, 3381
 Motorola DSP 56200
 Inmos A-100
 LSI Logic L64220, 64240

Case Studies: Designing with DSP Chips
- FIR & IIR Filters ■ Implementing FFTs
- Modulation ■ Adaptive Filters

4. USING MICROPROGRAMMABLE DEVICES

Designing DSP Architectures
- Bit-Slice ALU's and Pipelining
- Deriving Circuit Structure from Algorithms
- Optimizing the Bus & Memory Design
- Data & Coefficient Address Generation
- Branching & Subroutines ■ SISD vs. SIMD
- Timing, Propagation & Race Considerations

Developing the Microprogram
- The Microcode Word Structure
- The Programming & Data Flow Model
- Tools for Microprogram Development
- Building Mnemonics to Ease Programming

Multiplier/Accumulator-Based Systems
- Designing with TRW 1010J-type Chips
- Multi-ALU Pipeline & Parallel Designs
- DSP Microcode & Benchmarks
- ASIC Cells & Macros for DSP

Survey of Microprogrammable Devices
- ALU Architecture & Performance
- 2-Port Memories ■ Pipeline PROMs
- Sequencers ■ Address Generators
- Integer & Floating Point Slices:
 AMD 29500 & 29300 Families
 Analog Devices ADSP 1101, 8xxxx
 LSI Logic L64032, 64132
 TI 74AS8836, 8837
 Weitek WTL 3332
 Plessey PDSP 16112, 16316, 16330

Case Study: Real-Time Spectrum Analyzer
- Analysis of Processing Requirements
- Designing the ALU, Address Generators, & Sequencer Unit
- Developing the Microprogram

5. MICROPROCESSOR-BASED DESIGN

Considerations for 16 & 32-Bit Micros
- Power of Instruction Set & Registers
- Bandwidth & Processing Speed
- DSP Benchmarks for 16 & 32-Bit Chips

Case Study: Implementing an IIR Filter
- Detailed Microcomputer Hardware Design
- The Assembly Language Program

Learn by Doing

A unique feature of this course is that for each signal processor, the benchmarks of processing time and maximum sample rate are developed in class for:

- **FFTs ■ IIR & FIR Filters**
- **Convolution ■ Correlation**
- **Modulation**

Real-Time hardware designs utilizing DSP chips, microprogrammed bit-slices and 16-bit microprocessors are developed in-class and compared for:

- **Spectral Analysis**
- **Communications Systems**
- **Data Acquisition/Process Monitoring**

Through these case studies, participants acquire a working knowledge of practical design techniques for real-time signal and image processing hardware.

INTEGRATED COMPUTER SYSTEMS

ENROLL NOW!

Call Toll Free **(800) 421-8166** or

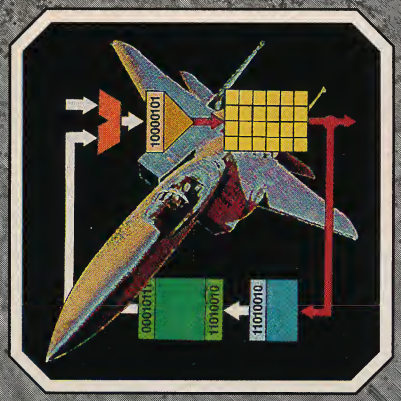
East Coast Office **(703) 847-6448**

West Coast Office **(213) 417-8888**

8805.UB.418

Digital Control Systems: Design Techniques & Tools

Course 418



4 Days

LOS ANGELES
June 14-17
TORONTO
June 21-24
WASHINGTON, D.C.
July 26-29
LOS ANGELES
August 30-September 2
WASHINGTON, D.C.
September 27-30
OTTAWA
October 18-21
SAN FRANCISCO
November 1-4
SAN DIEGO
December 13-16

Courses at Your Facility
Please Inquire

Course Benefits

Modern digital control systems are being designed and implemented in a rapidly expanding range of applications, due to the availability of powerful, reliable and inexpensive digital devices. This course provides the fundamental design methods, tools and techniques required for the successful implementation of digital control systems.

Throughout the course, a family of frequency variables is developed to establish a thorough understanding of mixed control systems, while providing expedient ways of converting analog systems to digital systems.

YOU WILL LEARN HOW TO:

- **Design Digital Control Systems for a Variety of Applications**
- **Select Digital Control Methods and Implement Them in Hardware and Software**
- **Utilize Digital Compensation to Stabilize Systems and Achieve Higher Levels of Performance**
- **Achieve Improved System Operation at Reduced Cost by Converting Analog-Based Systems to Modern Digital Systems**

WHO SHOULD ATTEND:

This course is valuable for circuit design, system and project engineers, programmers and technical managers who require a comprehensive understanding of the methods, tools and techniques for implementing modern digital control systems. A background in calculus, differential equations and LaPlace transforms is recommended.

Course Content

1. INTRODUCTION & OVERVIEW

Course Objectives
- Theory & Implementation
- Design Philosophy & Methodology
- Practical Real World Applications

Terminology & Basic Concepts
- Closed & Open Loop Systems
- Analog, Digital & Mixed Loops
- Transducers/Actuators ■ Plant Dynamics
- Time & Frequency Domain Concepts
- Sampling & Quantization

Digital vs. Analog Control
- The Computer as a Control Element
- Cost ■ Size ■ Flexibility ■ Speed
- Multiple Channels
- Advantages ■ Pitfalls ■ Tradeoffs
- Identifying Appropriate Applications

2. ANALOG SYSTEM DESIGN

Review of Analog Tools & Techniques
- Differential Equations
- Convolution & the Impulse Response
- Step & Ramp Responses ■ Bode Plots
- Laplace Transforms ■ Transfer Functions
- Frequency Response

Analog Closed Loop Systems
- Closed Loop Transfer Equations
- Root Locus Principles in the S-Plane
- Rise Time, Bandwidth, Overshoot
- PID Design ■ Lead/Lag Compensation
- Phaselock Systems ■ Inertial Systems
- Gain/Phase Margin ■ Settling Time
- Local Feedback Techniques

3. DIGITAL TOOLS & TECHNIQUES

How Digital & Analog Methods Relate
- The Z-Plane vs. the S-Plane
- Difference & Shift Equations
- Discrete Impulse Response & Convolution

Digital Data Sampling
- Ideal Waveform Sampling
- Sample & Hold Devices

Simulating Analog Systems Digitally
- Effects of Sample Rate
- Discrete Integration Techniques
- Comparison of Time Responses

Frequency Analysis for Digital Systems
- Fourier Analysis of Sampled Signals
- Relating LaPlace, Z & Transitional Variables
- Analog Simulation, Frequency Methods
- Comparing the S-, β-, W- and Z-Planes
- Z-Plane Analysis ■ Poles & Zeros
- Transfer Function ■ Stability

Using Root Locus for Digital Systems
- Usage in Design & Analysis
- Asymptotes ■ Breakaway Points
- Pole Zero Absorption ■ Gain Calculations
- Root Locus in the Z-, β- and W-Planes

Steady State Error Analysis
- Impulse, Step & Ramp Inputs
- Examples: Target Tracking, Speed Control

4. DIGITAL CONTROL SYSTEM DESIGN

Control Loop Fundamentals
- Single vs. Multi-Feedback Loops
- Developing Closed Loop Functions
- Destabilizing Effects of Sampling
- Handling the Zero-Order Hold
- Dealing with Computation Time Delay

Putting a Control System Together
- Design Philosophy ■ Component Modeling
- Sample Rate Selection

Improving Closed-Loop Performance
- System Performance Goals
- Approximations for Design Simplification
- Stabilization with Digital Compensation
- Root Locus as a Design Tool
- Local Feedback & PID Variations

5. IMPLEMENTING DIGITAL CONTROL SYSTEM HARDWARE & SOFTWARE

Hardware Components
- A/D & D/A ■ Sample & Hold Circuits
- Single Chip Micros
- Special Purpose Digital Processors

Real-Time Software
- Interrupt Driven Systems
- Software Control of Digital Devices
- Implementing Control Software

Finite Word-Length Effects
- A/D & Arithmetic Roundoff Noise Analysis
- Digital Filter & Closed-Loop Performance

Implementing a Digital Design
- Interfacing Digital Elements
- Parallel & Pipeline Techniques
- Timing & Synchronization
- Linearity Effects & Loop Performance

6. DEVELOPING AN APPLICATION

- Problem Definition
- Identifying Analog-Digital Interfaces
- Designing the Control Algorithm
- Detailed Hardware Design
- Control Software Implementation

7. SPECIAL TOPICS IN DIGITAL SYSTEMS

Digital Filters in Control Applications
- Noise & Carrier Suppression, Loop & Transducer Compensation
- Zero-, First- & Second-Order Hold Devices
- Digital Interpolation/Extrapolation
- FIR Techniques ■ IIR Techniques

Systems with Transport Delay
- Differential Difference Equations
- Frequency Plane Solutions
- Settling Time, Bandwidth, Stability
- Industrial Process Control Examples

CASE STUDIES

This course emphasizes practical applications of digital control techniques through real-world examples and case studies. Each analytical tool is developed from scratch rather than merely stated. This leads to some attractive approaches for designing high sample rate mixed systems, while providing clarity and instilling confidence in the application of these techniques to standard problems. Hardware and software implementations are discussed along with the timing and control issues of interfacing digital and analog components.

Case Studies include:

- **Microprocessor-Based Digital Filters**
- **Digital and Analog Speed and Positioning Systems**
- **Digital Phase Locking Systems**
- **Integrated Process Control Systems**

INTEGRATED COMPUTER SYSTEMS

ENROLL NOW!

Call Toll Free **(800) 421-8166** or

East Coast Office **(703) 847-6448**

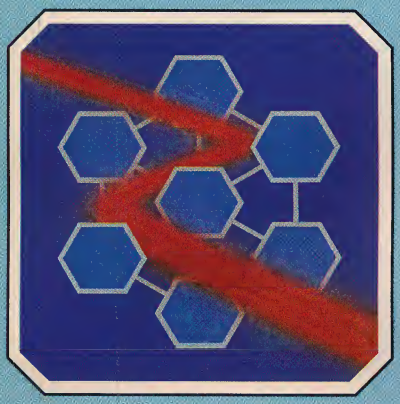
West Coast Office **(213) 417-8888**

Word-Slice is a trademark of Analog Devices, Inc.
OCCAM is the trademark of the INMOS Group of Companies.

8805 UB 169

Supercomputers, Hypercubes & High Performance Architectures

Course 169



4 Days

LOS ANGELES
June 7-10

TORONTO
June 14-17

BOSTON
June 21-24

WASHINGTON, D.C.
July 12-15

LOS ANGELES
September 20-23

BOSTON
October 25-28

OTTAWA
November 22-25

WASHINGTON, D.C.
December 6-9

Courses at Your Facility
Please Inquire

Course Benefits

The Architecture of High Performance Computer Systems is evolving rapidly, driven by increasing demands from users and the potential for creating "supercomputer" capabilities by linking multiple VLSI-based processors.

This course provides a comprehensive overview of the state-of-the-art in Modern Computer Architectures. Emphasis is on key practical design approaches and performance tradeoffs of alternative implementations. In-Class Case Studies clarify important design techniques and applications.

YOU WILL LEARN:

- **Key Concepts of High Performance Architecture Design**
- **Use of Pipelining, Vector & Array Processing**
- **The Ways in Which Cache Memory & Memory Management Affect Performance**
- **How Multiprocessor Techniques are Implemented in Super Minicomputers & "Hypercube-like" machines**
- **The Impact of Modern Computer Architectures on Operating Systems, Compilers, Languages and User Applications**
- **How to Estimate the Achievable Performance Improvements of Alternative Architectural Designs & Implementations**

WHO SHOULD ATTEND:

This course is valuable for hardware and software design and development engineers, systems architects, programmers, analysts and technical managers who are responsible for evaluating, configuring, programming or using high-performance computer systems.

Course Content

1. INTRODUCTION & OVERVIEW

Fundamental Concepts
- The von Neumann Machine
- Historical Evolution of Computer Architecture Families
- The Use of Concurrency: Applying Multiple Resources to Computational Problems
- Performance Measures: MIPS/MFLOPS/LIPS
- Evolving Hardware Technology: TTL to VLSI

Overview of Computer Architectures
- Classification of Computer Architecture
- Parallelism, Pipelining, Vector & Array Processing
- Uni- & Multiprocessors ■ Multicomputers
- Supercomputers, Mini Supers, Super Minis
- Hypercubes, Systolic Arrays, & Data Flow

High Performance Applications

2. CPU ARCHITECTURE, MEMORY & I/O SYSTEMS

CPU Design
- Accessing Operators & Operands
- Stack, Register, & Memory Approaches
- Applications of Addressing Modes
- Instruction Set Orthogonality
- Reduced vs. Complex Instruction Sets
- Microprogramming/Writable Control Store

Impacts of Hardware Technology
- High-Performance VLSI Microprocessors
- Arithmetic & Graphic Coprocessors
- Bit & Word Slice Components
- PLA's & Custom Gate Arrays

Memory Management Techniques
- Virtual Memory Concepts
- Segmentation & Paging Schemes
- Access Protection & Shared Memory
- On-Chip Memory Management

Designing High-Speed Memory
- Memory Bandwidth Determination
- Cache Memory Implementations
- Interleaving Memory ■ Multiported Memory
- Shared Memory ■ Local vs. Global Memory

I/O Subsystem Design
- Alternate Data Bus Designs
- I/O Bottlenecks ■ DMA Techniques
- Interrupt Mechanisms & Complexities
- VLSI Chips: I/O & Network Controllers

3. PIPELINING, VECTOR & ARRAY PROCESSING

Pipelining Techniques
- Throughput & Processor Utilization
- Pre-fetching and Overlapping Instructions
- Sequencing & Collision Prevention
- Examples: Amdahl 470-V8, M68020

Vector Oriented Machines
- Information Storage Mechanisms
- Concurrent Execution of Operations
- Operand Streaming & Pipelining
- Vector Processor Examples: Cray, IBM 3090

Array Processors - Single Instruction Multiple Data Stream (SIMD)
- Evolution of Array Processors
- Interconnecting Processing Elements
- Broadcast Buses, Meshes & Shuffle Nets
- Application "Fit" and Data Dependence
- Examples: Illiac IV, MPP, FPS-164

4. MULTIPROCESSOR SYSTEMS

Hardware Implementation & Architecture
- Connecting Processors, Memory, & I/O
- Data Transfer & Message Passing
- Loosely & Tightly Coupled Processors
- Symmetric vs. Asymmetric Systems
- Memory Organizations & Tradeoffs
- Overload, Deadly Embrace & Other Pitfalls

Multiprocessor Software
- Design of Executive & Scheduler
- Interprocessor Communications
- Process & Task Synchronization
- Resource Allocation ■ Load Balance

Data-Flow Machines
- Data Flow Graphs
- Data Driven Computing ■ Messages/Tokens
- Static vs. Dynamic Architectures
- Distributed/Asynchronous Control
- R&D Progress at MIT & Manchester

Multiprocessor Implementations
- Shared Memory Multiprocessors: Sequent Balance 8000
- Hypercubes - Intel iPSC
- INMOS Transputer, FPS T-Series, & the OCCAM Language
- Supercomputer: Cray XMP-4

5. PERFORMANCE EVALUATION

Benchmarking Techniques
- Identifying Appropriate Metrics
- Whetstone & Drystone Benchmarks
- Simulating Interactive Loads
- Pitfalls of Performance Evaluation

Evaluation/Selection Considerations
- Special vs. General Purpose Designs
- Tradeoffs - Cost, Performance, Compatibility, & Fault Tolerance
- Software Support & Ease of Programming

6. ONGOING R&D

- ICOT - 5th Generation Project
- European ESPRIT Program
- NYU/IBM: Ultracomputer Architecture
- Caltech Hypercubes ■ US Navy: S1
- Univ of Illinois: Cedar & Parafrase
- Thinking Machines: Connection Machine
- CMU: Warp Systolic Array ■ BBN Butterfly

7. FUTURE TRENDS

- Language Oriented Architectures
- Optical Computing and Storage
- GaAs & Supercooled Computers
- "Smart/Active" Memories

Design Case Studies

During In-Class Design Workshops, participants will develop **High Performance Architecture Design** solutions for the following applications:

TRANSPUTER CASE STUDY
- Partitioning the Problem
- Determining Interprocess Communications
- Utilizing the OCCAM Language
- Adding Processors to Improve Performance

CRAY CASE STUDY
- Identifying Parallelism
- Selecting the Processing Algorithms
- Using Data Storage Methods
- Employing Pipeline/Streaming Techniques

INTEGRATED COMPUTER SYSTEMS

ENROLL NOW!

Call Toll Free **(800) 421-8166** or

East Coast Office **(703) 847-6448**

West Coast Office **(213) 417-8888**

Effective Skills for Technical Managers

8805 UB 290

Course 290



4 Days

LOS ANGELES
June 7-10
WASHINGTON, D.C.
June 28-July 1
BOSTON
July 12-15
SAN DIEGO
July 19-22
SAN FRANCISCO
August 16-19
WASHINGTON, D.C.
August 23-26
LOS ANGELES
September 20-23
BOSTON
September 27-30
WASHINGTON, D.C.
November 1-4

Courses at Your Facility
Please Inquire

Course Benefits

When technical professionals are promoted to managerial positions, their success depends on how quickly they learn people-oriented management skills to supplement their prior technical knowledge. This unique "HOW-TO" course provides new, about-to-be, and experienced managers with the practical skills and tools needed to lead effective teams of technical professionals.

Participants will gain a working understanding of the key aspects of management . . . **leadership, delegation, motivation and communication** . . . as well as practical techniques for **planning, organizing, and controlling technical projects.**

YOU WILL LEARN HOW TO:

- **Develop the Perspective and Skills to Manage Effectively**
- **Attract, Motivate, Develop and Retain Competent People**
- **Build and Lead Productive, Committed Project Teams**
- **Plan, Organize and Control Technical Projects**
- **Delegate Tasks and Authority While Maintaining Control**
- **Communicate Effectively with All Levels of Management**
- **Manage Time Effectively in a "Multi-Tasking" Environment**

WHO SHOULD ATTEND:

This course is valuable for engineers, scientists, programmers, analysts, and other technical and data processing professionals who wish to refine and expand their management skills while increasing the motivation and productivity of their teams.

Course Content

1. INTRODUCTION AND OVERVIEW

Management Is a Profession
- The Benefits and Challenges of Management
- Management Skills Are Learned, Not Inherited
- How Your Technical Background Will Help You as a Manager

Focusing on Results and the People Who Help You Achieve Them
- Building and Leading a Team
- Assuring Effective Communication
- Planning and Controlling Projects
- Making Efficient Use of Time in a "Multi-Tasking" Environment

2. DEVELOPING LEADERSHIP SKILLS

Guiding Technical Personnel
- Developing Clear and Attainable Objectives
- Obtaining Commitment from the Team Members
- Maintaining a Productive Atmosphere
- Avoiding Indecision and Procrastination
- How to Keep Creative People Motivated

How Expectations Affect Results
- Setting Reasonable Expectations for Yourself and Your Team
- What the Team Expects from Its Leader

Delegating Effectively
- When, Why and How Much to Delegate
- How to Delegate Without Losing Control
- Assigning Authority and Accountability
- Tracking Progress and Measuring Performance
- Avoiding Reverse Delegation

Providing Constructive Feedback
- Shaping and Modifying Behavior
- Applying Positive vs. Negative Feedback
- Your Coaching and Counseling Roles

3. BUILDING A TEAM

Organizing the Structure
- Partitioning and Specifying Tasks
- Generating Meaningful Job Descriptions

How to Attract and Retain Competent People
- Conducting Performance Appraisals that Improve Morale and Productivity
- Developing Your Staff's Potential

Intra- and Inter-Team Coordination
- Creating Systems and Procedures
- Negotiating and Resolving Conflicts
- Adjusting to Changing Objectives
- Operating in a Political Environment

4. EFFECTIVE COMMUNICATION SKILLS

Communication Styles and Channels
- Communicating with Subordinates, Peers and Higher Management
- When to Use Verbal vs. Written Communication
- The Grapevine: How to Deal with It

Verbal Communication Skills
- How to Listen Actively
- Providing Feedback
- Aggressive Inquiry: How to Get the Facts

5. PLANNING AND CONTROLLING PROJECTS

Developing a Project Plan
- Establishing the Project Specification
- Translating the Project "Mission" into Specific, Concrete Goals
- Setting Meaningful and Measurable Milestones
- Developing Contingency Plans
- Guidelines and Checklists for Planning

Estimating Time and Costs
- Creating the Initial Estimates
- Using Work Breakdown Structures
- Updating and Refining Estimates

Controlling the Project
- Implementing Tools to Monitor Progress
- "90% Complete" and Other Pitfalls
- PERT, CPM and Other Analytical Tools
- Developing Time-Phased Activity Plans

6. MANAGING TIME

Basic Time Management Techniques
- Common "Time Traps" & How to Avoid Them
- Focusing on High Payoff Activities
- Handling Conflicting Priorities
- Knowing When and How to Say "No"
- Prime and Block Time Scheduling

Optimizing Team Efficiency
- Organizing Activities to Minimize Interruptions
- How to Conduct Productive Meetings
- Implementing Prioritized "To-Do" Lists
- Setting Weekly Goals and Schedules

7. AFTER THE COURSE: CONTINUED GROWTH

- A Specific On-the-Job Program for Applying Skills Learned in the Course
- Maintaining Momentum in the Face of Real-World Pressures
- A Strategic Plan for Personal Development

Special Course Features

THE WORKSHOP FORMAT

The workshop format is used extensively throughout the course. Case studies, team activities and in-class exercises provide participants with first-hand experience in applying the management tools presented.

VIDEO PRESENTATIONS

Selected video tapes illustrate the impact of appropriate and inappropriate management techniques in real-world situations.

MANAGEMENT PROCEDURES HANDBOOK

Contains key forms and procedures for planning, organizing, estimating and scheduling technical projects.

ON-THE-JOB EXERCISE WORKBOOK

This unique workbook contains a carefully designed sequence of "direct action" exercises for use after returning from the course. These exercises reinforce the participant's understanding of key management techniques by providing step-by-step guidance for applying them on the job.

INTEGRATED COMPUTER SYSTEMS

ENROLL NOW!

Call Toll Free **(800) 421-8166** or

East Coast Office **(703) 847-6448**

West Coast Office **(213) 417-8888**

8805 UB 340

Software Project Management Tools & Techniques

DEMO LABS

Course 340



4 Days

PALO ALTO
June 7-10
BOSTON
June 21-24
TORONTO
July 5-8
WASHINGTON, D.C.
July 12-15
SAN DIEGO
August 2-5
OTTAWA
August 9-12
WASHINGTON, D.C.
September 13-16
LOS ANGELES
October 25-28
WASHINGTON, D.C.
November 15-18

Courses at Your Facility
Please Inquire

Course Benefits

This course develops and improves the skills required to successfully plan and manage software development efforts. It provides an overview of the development process, presents an approach that has been proven to work, and provides a **comprehensive and practical set of management tools that can be implemented immediately upon return from the course.**

YOU WILL LEARN HOW TO:

- **Plan, Organize, Monitor, and Control Software Projects**
- **Specify Every Element of a Workable Project Plan**
- **Produce Reliable Estimates of Cost, Schedule, and Manpower**
- **Use Documentation to Control and Support the Project**
- **Develop Quality Software and Reduce Maintenance Headaches**
- **Implement a Test Strategy That Will Insure a Working Product**
- **Increase Productivity and Cut Costs**

WHO SHOULD ATTEND:

This course is valuable for software and hardware engineers, systems & applications programmers, systems analysts and other technical and data processing professionals who either currently manage or are about to manage software development projects.

Course Content

1. INTRODUCTION AND OVERVIEW

Understanding the Development Phases
- Defining Software Requirements
- The Design Phase
- The Implementation Phase
- SW/HW Integration and Test Phase
- Product Support and Maintenance

Key Project Management Skills
- Defining the Deliverables
- Preparing a Good Project Plan
- Estimating Time and Costs
- Implementing Management Controls

2. DEVELOPING THE REQUIREMENTS AND DESIGN SPECIFICATIONS

Using the Requirements Document as a Communications Device
- Determining and Codifying Requirements
- Allocating Requirements to Subsystems
- Producing Concrete Specifications
- Using Automated Tools

How to Conduct the Design Phase
- Applying Top-Down Methodology
- When and How to Conduct Walk-Throughs
- Designing Software for Flexibility and Portability

3. ESTIMATING THE PROJECT COST AND SCHEDULE

Practical Estimating Techniques
- Creating the Initial Estimate
- The Use of Resource Models
- Using Work-Breakdown Structures
- Applying Statistical Methods to Revise Estimates and Continually Reduce Error

Estimating Program and Memory Size
- Lines of Code — The Starting Point
- Asking the Right Questions to Develop Good Estimates
- Calculating the Expected Object Code

Estimating the Development Effort
- Translating System Size into Man-Months
- Avoiding Common Estimating Errors
- How to Sell the Estimate
- Defending Against Creeping Commitments

4. PLANNING THE PROJECT

A Planning Methodology
- Setting Clear Overall Goals
- Identifying Responsibility & Accountability
- Subdividing the Project into Tasks

Developing the Project Plan
- Setting Objectives and Defining Measurable Milestones
- Converting Objectives into an Action Plan
- Using PERT & Other Scheduling Methods
- Identifying the Key Resource Requirements and Potential Bottlenecks
- Setting Reporting Formats and Schedules

Organizing the Project Team
- Successful Team Structures: Chief Programmer Teams, Matrix Management, Functional Organization, "Tiger Teams"
- Insuring Good Communication Channels

5. GUIDING THE IMPLEMENTATION

Minimizing Implementation Time & Cost
- Using Structured Programming Methods
- Impact of Programming Language
- Reviewing the Code
- The "Programmer's Workbench" Concept
- Productivity Guides

Unit Testing
- Assuring Early Testing of Each Module
- Avoiding the "90% Complete" Pitfall

6. MANAGING THE INTEGRATION & TEST

A Practical Testing Methodology
- Static and Dynamic Testing Methods
- What to Verify After Each Phase
- Test Specification and Procedures
- Test Scheduling, Reporting and Control

Coordinating SW/HW Integration/Test
- How to Achieve Early Error Detection
- The Build/Test Cycle
- Regression Testing ■ System Testing
- Integration and Test Bottlenecks

7. CONTROLLING SOFTWARE CHANGES

The Software/Hardware Baseline Concept
- Identifying the Configuration
- SW Parts List ■ SW Document List
- Change Control Procedures

Documentation: The Practical Approach
- Global, Top-Level and Detail Documentation
- Maintenance and User Documentation
- Tools for Automating the Documentation

DEMO LABS

Participants apply project management tools and techniques by developing a comprehensive project plan that includes:

- **Specifying the Deliverables**
- **Analyzing Requirements**
- **Developing Work-Breakdown Structures**
- **Generating PERT Task Networks and GANTT Critical Path Schedules**
- **Estimating Manpower, Schedule & Cost**

Participants will see these functions demonstrated in-class on a Computer-Based Project Management System.



Software Development Procedures Manual

In addition to the Course Notes, each participant receives the "Software Development Procedures Manual" with comprehensive procedures and organizational tools for use in managing their projects.

INTEGRATED COMPUTER SYSTEMS

ENROLL NOW!

Call Toll Free **(800) 421-8166** or

East Coast Office **(703) 847-6448**

West Coast Office **(213) 417-8888**

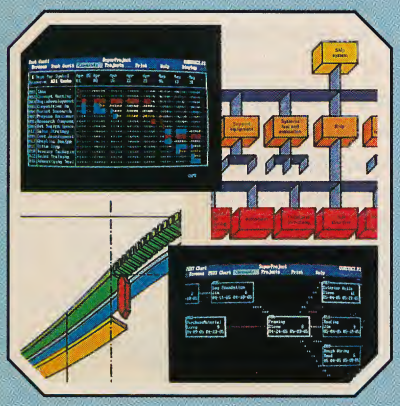
8805 UB 296

Project Management: Planning, Scheduling, Tracking & Controlling

HANDS-ON

Course 296

NEW!



4 Days

LOS ANGELES
August 16-19
OTTAWA
September 13-16
WASHINGTON, D.C.
September 20-23
TORONTO
October 25-28
SAN DIEGO
November 1-4
MONTREAL
November 15-18
LOS ANGELES
December 13-16
WASHINGTON, D.C.
January 10-13

Courses at Your Facility
Please Inquire

Course Benefits

A successful Project Manager is one who knows both WHAT has to be done, and HOW to do it. This course provides participants with proven **"how-to" concepts and procedures for organizing, planning, scheduling, monitoring and controlling projects.** Comprehensive, in-class workshops provide hands-on experience with manual and PC-based tools and techniques which are essential to successful project management.

Participants are supplied with a set of guidelines, forms, and checklists which are ready for use immediately upon return to their work.

YOU WILL LEARN HOW TO:

- **Fully Define Your Job as a Project Manager in Specific Terms That Will Systematically Guide Your Actions**
- **Develop "Accountability Contracts" for Your Subordinates**
- **Establish Effective Inter and Intra-Group Communication**
- **Use PC-based and Manual Tools for Creating Schedules, Budgets, Work Breakdown Structures, and PERT Networks**
- **Implement and Manage Monitoring Tools and Controls to Keep You Fully in Command of the Project**
- **Avoid the Pitfalls of "Creeping Commitments"**

WHO SHOULD ATTEND:

This course is valuable for those who need to know how to successfully manage and control projects. Emphasis is on the practical, hands-on, "how-to" application of both manual and PC-based tools and techniques. The course is appropriate for both new and experienced project managers.

Course Content

1. INTRODUCTION AND OVERVIEW

Fundamental Concepts and Goals
- Basic Tasks of Management
- Defining and Delimiting a Project
- Controlling Resources & Schedules
- Identifying Obstacles to Success

Tools of the Trade
- Defining Your Task as a Project Manager
- Planning the Project, a Pragmatic Approach
- Analyzing Task Flow: Manual and PC Tools
- Budgeting and Cost Control
- Scheduling and Schedule Tracking
- Establishing Change Control

Adapting to the Project Manager's Role
- The Critical First Step: Accepting Ownership and Responsibility
- The Internalization Process
- Thinking in Terms of Actions
- Leading the Project Team
- Using a Checklist of Management Functions

2. SUCCESSFUL PROJECT PLANNING

Specifying the Project
- Defining Project Goals & Deliverables
- Estimating Time, Cost and Resources
- Dealing With Imposed, Unrealistic Goals

Planning the Project
- Defining Project Life Cycle Phases
- Using Decision Trees to Drive Actions
- Identifying Significant Milestones
- Structuring a Task List

PC Tools for Project Management
- Using PERT/CPM & Gantt/Bar Chart Tools
- Allocating & Budgeting Using Spreadsheets
- Applying Project Management Packages, Database Managers and Word Processors

Producing the Project Plan
- Establishing a Baseline Plan
- Using Manual and PC-Based Project Management Tools to Produce a Plan
- Planning the Work Flow
- Developing Schedules and Budgets
- Using Task (Precedence) Networks: PERT
- Performing Critical Path Analysis and Schedule Tracking
- What-if Scenarios & Contingency Planning
- A Practical Approach to Risk Analysis

3. HOW TO ORGANIZE THE PROJECT AND GET YOUR TEAM'S COMMITMENT

Building and Leading a Dedicated Team
- Establishing a "Work Contract" Mind Set
- Communicating the Project Plan
- Transferring Plan Ownership to the Team
- Negotiating for Resources & Commitments
- Getting Help From People You Don't Manage
- Applying the Principles of Participative Management

Allocating Resources & Assigning Tasks
- Establishing Team Work Breakdown Structures
- Loading and Leveling of Resources
- Using Team-oriented Task Definition: MBO (Management By Objectives) That Works
- Assessing Team Member Performance

Project Structure & Organization Styles
- Structuring the Project Organization
- Using Delegation & Power
- Applying Persuasion & Influence
- Resolving Conflicts
- The Role of the Project Manager in Multi-project & Matrix Organizations

Administrative & Computer Support
- Determining Needs for Administrative Support
- Selecting Manual vs. Automated Support Tools

4. IMPLEMENTING PROJECT CONTROLS

Designing Effective Project Controls
- Understanding the Goals of Project Controls
- Specifying the Control Requirements
- Defining Project-Specific Controls

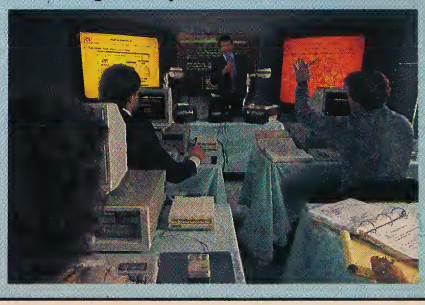
Tracking Performance Against the Plan
- Selecting the Parameters to Track: Task Progress, Cost Trends, Earned Value Reporting
- Establishing Change Control and Configuration Management
- Running Effective Meetings: Audits, Status, Design Reviews

Correcting & Adjusting the Plan
- Monitoring & Reporting Progress
- Measuring Variance from the Plan
- Controlling External Dependencies
- Modifying the Baseline Plan
- Concluding the Project

HANDS-ON

Participants in this course gain pragmatic, "how-to" skills through a series of **interactive small group workshops** in which participants gain expertise in implementing the principles presented in class. Participants perform the steps a Project Manager typically follows in planning, running and completing a project, **using both manual and PC-based tools.**

1. **Define Project Objectives and Deliverables**
2. **Generate an Internal Workscope "Contract"**
3. **Create the Work Breakdown Structure and Project Plan**
4. **Develop Explicit Work Objectives for Team Members**
5. **Produce the Implementation Plan Using PC-Based Tools**
6. **Implement a Project Control Package**
7. **Maintain a Change Control File**
8. **Revise the Plan to Meet Changed Requirements**



INTEGRATED COMPUTER SYSTEMS

ENROLL NOW!

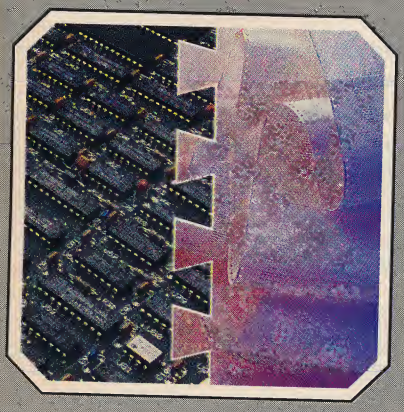
Call Toll Free **(800) 421-8166** or

East Coast Office **(703) 847-6448**

West Coast Office **(213) 417-8888**

System Engineering for Integrated Hardware/ Software Applications

Course 348 **NEW!**



4 Days

LOS ANGELES
June 21-24
SAN DIEGO
July 12-15
WASHINGTON, D.C.
July 19-22
SAN FRANCISCO
August 23-26
WASHINGTON, D.C.
September 20-23
LOS ANGELES
September 27-30
SAN DIEGO
October 18-21
BOSTON
November 15-18

Courses at Your Facility
Please Inquire

Course Benefits

8805 UB 348

The System Engineering Methods presented in this course facilitate the orderly development of clear requirements and result in the implementation of reliable, cost-effective Hardware/Software products, on time and within budget. This course provides a working knowledge of the interdisciplinary techniques needed to successfully develop and integrate HW/SW systems for both very large and small projects. **Participants will learn practical skills to define systems, perform trade-offs, allocate requirements, budget resources, manage HW/SW configurations, and verify, integrate and test the product.**

YOU WILL LEARN HOW TO:

- **Translate Objectives into HW/SW Requirements**
- **Formulate a System Architecture and Partition Requirements**
- **Plan and Execute Interdisciplinary Trade-off Analyses**
- **Allocate Performance Requirements to System Elements**
- **Budget and Manage System Technical Resource Margins**
- **Develop HW/SW Integration and Verification Plans**
- **Adjust to Changes in Requirements, Schedules, Funding**

WHO SHOULD ATTEND:

This course is valuable for Hardware and Software engineers, systems analysts, and managers who participate in the development of hardware/software systems. It is of value both for recent graduates, and for experienced engineers who have focused on a single technical discipline and wish to expand their perspective to practical system engineering.

Course Content

1. INTRODUCTION AND OVERVIEW

What is System Engineering?
- Conceptual Design of HW/SW Systems
- Allocating Functions to System Elements
- Defining HW/SW Requirements & Trade-offs
- Budgeting Resources, Performance, Margins
- Technical HW/SW Coordination
- Configuration Management
- Interdisciplinary HW/SW Team Leadership
- Design, Development and Implementation
- Applies to Both Large (e.g. Satellite) and Small (e.g. Bank ATM) Projects

The System Life Cycle
- Life Cycle Phases and Activities
- The System Engineer's Role at Each Phase
- Reviews ■ Test Considerations
- Incremental Development
- Life Cycle Costing

2. DEVELOPING THE CONCEPTUAL DESIGN

Defining the Product Context
- Customer Objectives
- Project Requirements and Constraints
- Conducting Option/Trade-off Analyses
- Understanding the "State of The Art"
- Technical Risk Identification/Assessment

The System Conceptual Design Phase
- Defining and Using Operations Scenarios
- Formulating Candidate HW/SW Architectures
- Defining a Conceptual Baseline: Product, Cost, Schedule

System Delivery Criteria and Schedules
- Defining Deliverables
- Scheduling Incremental Deliveries
- Acceptance Criteria and Procedures

3. DEVELOPING SYSTEM REQUIREMENTS

Establishing Quantifiable Requirements
- Definition of Requirements
- Attributes of Good Requirements
- Requirements Hierarchy and Traceability
- Functions ■ Performance ■ Interfaces
- Maintainability ■ Reliability ■ Safety
- Fault Tolerance ■ Human Factors

The Requirements Development Process
- Generating System Requirements
- Requirements Organization, Evaluation and Validation
- Requirements Design Engineering Tools

Requirements Analysis and Flowdown
- Establishing a Functional Baseline
- Partitioning Criteria
- The System Preliminary Design Review (PDR)
- Defining The Allocation Baseline
- Allocating Functions to HW & SW Subsystems
- Scoping and Allocating Resource Margins
- Partitioning Hardware and Software Tasks
- Subsystem Interface Definition and Control
- Concurrent HW/SW Development and Trade-offs
- Documenting the Design
- The System Critical Design Review (CDR)

4. PRODUCING A FEASIBLE DESIGN

Establishing Optional Designs
- Developing/Iterating Requirements
- Performing Design Trade Studies
- Assessing Design and Resource Impacts

System Verification Requirements
- Design Verification Considerations
- Verification Techniques and Strategies

Establishing the Detailed Design
- Expanding the Level of Detail
- Design for Embedded Computer Systems
- Evaluating the Design
- Responding to Changes

Modeling the Proposed System
- Tools, Capabilities, Limitations
- Analyzing vs. Simulating vs. Prototyping
- Using Models

5. IMPLEMENTING THE DESIGN

Establishing a Product Baseline
- Building HW Assemblies and SW Modules
- Auditing Development and Resource Utilization
- Concurrent HW/SW/Procedure Implementation

Design Verification
- Verifying the Product
- Preparing for System Integration
- The Subsystem Delivery Review

6. SYSTEM INTEGRATION AND VERIFICATION

Defining an Integration Test Strategy
- Estimating Resource Requirements
- Handling Verification Requirements
- Testing Procedures ■ Operations

Managing the Integration Process
- The Matrix of Supporting Elements
- Coordinating and Managing Changes

Integrating and Verifying the Product
- Integrating the HW/SW Subsystems
- Key Issues In Subsystem Integration
- Preparing for Testing
- Organizing Team Support
- Monitoring the Test Activity
- Testing Failure Tolerance
- Post Test Activities
- Verifying Incremental Deliveries
- Alpha and Beta Site Testing
- Formal Reviews and Review Documentation
- Failure Recovery and Error Correction

Planning and Scheduling Delivery
- What are the System Deliverables?
- Defining Capability Constraints and Liens
- The Release Description Document
- Operations and Maintenance
- Training Operations Personnel
- Adding System Enhancements

7. COURSE REVIEW AND SUMMARY

- Keys to Successful System Engineering
- System Engineering Axioms

Course Workshops

In-class workshops are used to illustrate the principles presented. Participants will:

- **Develop a Project Context**
- **Define a Conceptual Baseline**
- **Select a HW/SW Architecture**
- **Solve a Resource Margin Problem**
- **Respond to a Major Baseline Change/Error Budget Modification**
- **Plan a System Verification Strategy**

INTEGRATED COMPUTER SYSTEMS

ENROLL NOW!

Call Toll Free **(800) 421-8166** or

East Coast Office **(703) 847-6448**

West Coast Office **(213) 417-8888**

8805UB

Security for Computer and Communication Systems

Course 380



4 Days

WASHINGTON, D.C.
May 24-27
TORONTO
June 21-24
LOS ANGELES
June 28-July 1
BOSTON
July 19-22
WASHINGTON, D.C.
August 9-12
OTTAWA
September 27-30
LOS ANGELES
November 29-December 2
WASHINGTON, D.C.
December 13-16
MONTREAL
January 24-27

Courses at Your Facility
Please Inquire

8805 UB 380

Course Benefits

Software and databases that are vital components of an organization's information and operational procedures are assets of critical value. Threats to these assets arise from human error, natural disasters, surreptitious explorations, and from the malicious intent to penetrate a system and acquire or corrupt information illegally.

This course provides participants with a solid understanding of the ways in which the security of a computer system and its communication channels may be breached, together with **detailed procedures, tools and techniques for implementing sound Computer and Communications Security systems** for protection against data loss, corruption and intrusion.

YOU WILL LEARN HOW TO:

- **Recognize Signs that Warn of an Attack on Your System**
- **Use Formal Tools and Techniques to Evaluate Risks**
- **Identify Hardware and Software Vulnerabilities**
- **Detect Procedural and Personnel Related Vulnerabilities**
- **Implement Low-Cost Security Measures for PC Networks**
- **Utilize Encryption, Authentication and One-way Functions**
- **Develop and Implement a Security Strategy for Your Organization**

WHO SHOULD ATTEND:

This course is valuable for programmers, systems analysts, MIS professionals, systems designers, managers and security professionals who share or direct the responsibility for designing, implementing and managing computer and telecommunication security programs.

Course Content

1. INTRODUCTION & OVERVIEW

Evolution of Security Requirements
- Batch Computer Systems
- On-Line Systems with Local Access
- Remote Dial-up to On-Line Systems
- Wide Area Computer Networks
- Local Networks ■ Personal Computers

Assets, Threats & Risks
- Identifying & Valuing Assets
- Asset Value & the Cost of Security
- Defining Sources of Threat
- The Definition of Risk

Key Aspects of Computer Security
- Information & Tangible Properties
- Protection of Data, Software & Hardware
- Employees & Consultants
- Natural Disasters
- Outside Attackers ■ Terrorists

2. SECURITY SYSTEM FUNDAMENTALS

Preventing & Detecting Loss & Attacks
- Controlling System Access
- Monitoring to Detect Unusual Change
- Techniques for Detecting Intrusion Early

Recovering From Loss
- Creating Response/Contingency Plans
- Errors ■ Disaster ■ Security Breaches
- Dangers of Retrofitted Solutions

Analyzing & Managing Risk
- Identifying Potential Threats
- Determining System Vulnerabilities
- Models for Estimating Expected Losses
- Countermeasures Cost/Benefit Analysis
- The Crucial Role of Management

3. SECURITY & THE LAW

Impact of Government Regulations
- Application: Civilian or Military?
- Special Requirements of Classified Data
- Discretionary vs. Mandatory Policies
- Legal Responsibilities & Obligations

Survey of Relevant Legislation
- NSDD 145 ■ DoD Security Initiative
- Financial & Privacy Regulations
- Britain, France, Canada, Sweden, Others

Auditing & Audit Trails
- Who Did What to Whom - & When?
- Logon, User & Account Audit Trails
- Records of File & Device Access
- Data Modification & Retrieval History

4. COMPUTER HARDWARE SECURITY

Physical Security Measures
- Locate Hardware for Maximum Security
- Implementing Physical Access Control
- Card Access Control & "Smart" Cards
- Voice & Finger Prints, Retina Scans
- Effective Use of Seals

Backing Up Your Hardware
- How Long Can You Function Without It?
- Dual Systems & Hot-Sites
- Software Backups & Off-Site Archives
- Protecting Against Disaster

5. SOFTWARE & DATA SECURITY

Software Risk Assessment
- System vs. Application Software
- The Software Trojan Horse
- Trap Doors ■ Software Viruses
- Using Software Engineering Principles: The Visibility of Structured Code

Logical Computer Access Controls
- Passwords and Password Techniques
- Security Code Generators
- File Access Control & Passwords

Formal Security Models
- The Trusted Computing Base (TCB)
- Operating System Security Kernels
- The Bell-LaPadula Model
- Information Flow & Access Matrix Models
- Limitations of Formal Models

Multi-Level Secure Operating Systems
- Honeywell SCOMP (KSOS-6)
- UCLA Data Secure UNIX (DSU)
- IBM KVM/370 ■ DEC Approaches

Unique Considerations for Databases
- Securing Multilevel Databases
- Access Control of Database Views
- Preventing Inferential Penetration
- Monitoring for Trojan Horses

6. SECURITY FOR COMPUTER NETWORKS

Communicating Over Networks
- Computer Security vs. Network Security
- Communication Links & Channels
- Emanations Security & TEMPEST
- Modem Intercept and Call-Back Devices
- Telephone Taps ■ Microwave Intercept

Overview of Cryptographic Techniques
- Fundamental Cryptographic Principles
- The NBS Data Encryption Standard (DES)
- RSA & Other Public-Key Approaches
- Available Encryption Hardware & Software

Security in OSI Networks
- Security at the Data Link Level
- Session Level Security
- Application Level End-to-End Security
- International Data Flow

Authentication
- Cryptographic Authentication
- Digital Signatures
- Arbitration Protocols
- Error Detecting and Correcting Codes

The Problem of "Hackers"
- Who Are the Hackers?
- Hacker Networks & Bulletin Boards
- Approaches for Handling Hackers
- "Exhaustive" Attacks by PCs

PCs & PC Networks
- Protecting the PC from Its User
- Protecting Other Systems from the PC
- Access Control & Authentication
- Problems of Decentralized Control

7. IMPLEMENTING SECURITY POLICIES

Putting the Policy in Place
- System Analysis & Design
- Distributed vs. Centralized Control
- Operational Procedures
- Accountability & Auditing Procedures
- Indoctrination & Training

In-Class Exercises

Participants will gain practical experience by using tools and techniques to establish system asset value, analyze risks, define security procedures, estimate security costs, and produce a security plan for a typical organization.

INTEGRATED COMPUTER SYSTEMS

ENROLL NOW!

Call Toll Free **(800) 421-8166** or East Coast Office **(703) 847-6448** West Coast Office **(213) 417-8888**

COURSE AUTHORS AND INSTRUCTORS*

■ μP Software, Hardware & Interfacing p. 2

Mr. Martin Shinn has over 10 years of experience in the design and implementation of 8, 16 and 32-bit μP-based board products. He has developed both the hardware and software for a variety of applications ranging from traffic controllers to high speed telecommunications and real-time control systems.

Other Faculty: Mr. W. Brown, Mr. A. Carbone, Mr. C. Davidson, Mr. E. Dillingham, Mr. R. Krook, Dr. G. Starr and Mr. C. Wagar.

■ μP Troubleshooting p. 3

Mr. Stephen Leyland has designed hardware and software for a variety of μP families. His background includes digital signal processing, military communication systems and ATE applications. He has extensive experience in the application of microcomputer troubleshooting techniques and equipment.

Other Faculty: Mr. E. Dillingham, Mr. R. Krook, Mr. R. Morrison, Mr. M. Shinn, Mr. C. Wagar and Dr. A. Wilcox.

■ 16 & 32-Bit μP Systems p. 4

Mr. Philip Lieberman
Mr. Robert Brians has over 12 years of experience in the development for μP-based systems. His practical experience includes work with 8, 16 and 32-bit processors for applications ranging from multi-tasking, real-time systems to advanced μP-based equipment for automated test and fault isolation.

Other Faculty: Mr. W. Brown, Mr. E. Dillingham, Mr. R. Harvey and Mr. R. Krook.

■ Digital Signal Processing Techniques p. 5

Mr. William Schaff has over 12 years of experience in radar target identification techniques, sonar signal processing, systems design and spectrum analysis. As co-founder of Mathematical Systems Design, Inc., his responsibilities include the management of projects in areas such as high-speed digital filters, array processor peripherals, and military FFT processors.

Other Faculty: Dr. D. Bukofzer, Mr. D. Fink, Mr. D. Freberg, Mr. H. Gaggioni, Mr. R. Karwoski and Dr. M. Piovoso.

■ Image Processing & Machine Vision p. 6

Dr. Werner Frei has 12 years of experience in Digital Image Processing and Pattern Recognition, including the development of real-time systems for image enhancement and classification, medical imaging and robot vision.

Other Faculty: Dr. I. Abdou, Dr. E. Angel, Dr. B. Baxter, Dr. J. Brayer, Dr. M. Cannon, Dr. J. Duncan, Dr. N. Griswold, Mr. B. Grush, Dr. T. Huang, Ms. S. Kemsley, Dr. S. Rajala, Dr. W. Snyder, Dr. J. Trussell and Dr. J. Youngberg.

■ Digital Signal Processing Hardware p. 7

Mr. Dennis Fink has over 10 years of experience in the design of microprogrammed processors, microprocessor-based systems and special purpose analog/digital hardware for signal and image processing. He has implemented FFT processors, high-speed digital filters, and array processor peripherals for applications in communications and process control.

Other Faculty: Dr. C. Armstrong, Mr. D. Freberg, Mr. H. Gaggioni, Dr. H. Pollard and Mr. W. Schaff.

■ Digital Control System Tools p. 8

Mr. Richard Karwoski has been a consultant to major corporations for over 12 years on the system design and detailed hardware/software implementation of a wide variety of analog and digital signal processing and control systems. He has designed and developed systems for industrial process control, shipboard fire control, inertial guidance, and digital filter systems.

Other Faculty: Dr. K. Hagen and Dr. M. Piovoso.

■ Supercomputer Architectures p. 9

Dr. Howard Pollard has 14 years of experience in the design and implementation of digital systems and high performance computer architectures. He is currently involved in the design and implementation of a 32-bit microcomputer-based Hypercube machine.

Other Faculty: Dr. C. Armstrong, Mr. C. Davidson and Mr. D. Freberg.

■ Skills for Technical Managers p. 10

■ Software Project Management p. 11

■ Project Planning & Control p. 12

Mr. Raymond Nienburg has 25 years of experience at IBM, Control Data, and MSI Data in the technical management of software/hardware projects including information systems, communication networks, defense systems, and commercial products, His experience includes FORTRAN, PASCAL, and Assembly Language on hardware ranging from embedded microprocessors to large computer networks.

Other Faculty: Ms. P. Hurst, Dr. T. McDonald, Mr. J. Mimmack, Mr. J. Rivenell, Mr. P. Schenck, Mr. F. Stern-Montagny, Mr. J. Weigant, Mr. D. Wessels and Ms. L. Whitley.

■ System Engineering p. 13

Mr. Brian Larman has been active in the field of System Engineering for over 25 years, and is currently the Project Software System Manager for a Jet Propulsion Laboratory orbiting spacecraft project. Mr. Larman brings a broad range of practical application experience to the system engineering discipline with a background which encompasses project conceptual design, system specification, hardware/software subsystem integration and testing, and control center and field station operations.

Other Faculty: Mr. G. Katz.

■ Security: Computers & Communication p. 14

Mr. David Snow has almost 20 years of experience in the design, implementation, and maintenance of secure computer and communications systems. His practical background has included work on secure operating systems and multi-level secure networks, TEMPEST analysis, and risk analysis methodologies. Mr. Snow has analyzed the security of dozens of commercial and military computer and communication systems, and was principle security architect of the first truly multi-level secure communications system.

Other Faculty: Dr. J. Cole, Dr. A. Cooper, Ms. D. Helle, Mr. J. Katz and Mr. J. Simon.

*Each course is taught by the author or one of a team of highly skilled instructors.



Learn from Experts

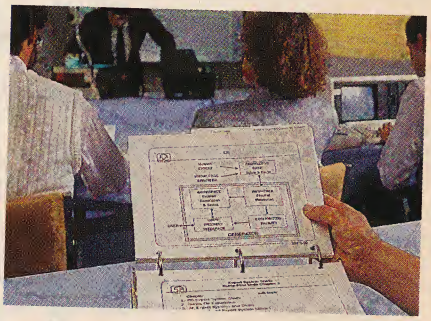
These courses are developed and taught by technical professionals . . . for technical professionals. You'll benefit from a well-organized presentation that emphasizes the practical application of state-of-the-art technology.



Effective Methods

Dual overhead projectors insure rapid communication. Where appropriate, slides, films, videotapes and live demonstrations augment the presentation.

In many courses, computer-based training with hands-on exercises, lab sessions and small team workshops provide concrete "Learn-by-Doing" experience.



Comprehensive Materials

Comprehensive Course Notes replicate each projected transparency, permitting full concentration on what is being taught.

The "State-of-the-Art Supplemental Readings" amplify key course topics, and provide a valuable "after course" reference.

PRODUCTIVITY THROUGH EDUCATION®

13 COURSES ON ENGINEERING & MANAGEMENT

HOW TO ENROLL

PHONE:

TOLL FREE: (800) 421-8166

INTEGRATED COMPUTER SYSTEMS

East Coast Office
(703) 847-6448

West Coast Office
(213) 417-8888

WRITE:

8000 Towers Crescent Dr.
Suite 300
Vienna, VA 22180
Fax: (703) 847-6319

5800 Hannum Ave.
P.O. Box 3614
Culver City, CA 90231-3614
Telex: 4995168 ICS LA
Fax: (213) 410-2952

Phone Today to Reserve Your Place:
Class sizes are limited and classes are often filled to capacity. We recommend that you enroll early.

Immediate Confirmation:
You will receive immediate confirmation of enrollment and course location details by phone (or telex) and subsequently by mail. You can reserve your place without obligation. There is no cancellation fee if you notify us at least two weeks before the course. If a course is cancelled for any reason, you will be notified at least two weeks in advance.

$100 Prepayment Discount:
While our long-standing policy is that payment is not required until after you attend the course, we are pleased to offer a $100 discount for prepayment. Payment may be made by check or major credit card: VISA MasterCard AMERICAN EXPRESS

$100 Multiple Enrollment Discount:
A $100 discount for each participant is provided when 2 or more from the same organization enroll and are invoiced together. Each gets the discount even if he or she enrolls for different courses or different sessions of the same course. PLEASE NOTE: This discount is available *in addition* to the $100 prepayment discount.

Please take advantage of these discounts and call today to enroll.

Course Fee:
The course fee includes a comprehensive set of course materials, and morning and afternoon refreshment breaks each day. These course materials include a detailed five-level set of course notes that parallels the presentation, case studies, all workshop session exercises, and a set of State-of-the-Art Reprints.

Course	Title	Length	Discount Prepaid Fee	Standard Fee
142:	Microprocessor Troubleshooting HANDS-ON	4 Days	$1295	$1395
161:	Microprocessor Software, Hardware & Interfacing HANDS-ON	4 Days	$1195	$1295
166:	16 & 32-Bit Microprocessor System Design	4 Days	$1195	$1295
169:	Supercomputers, Hypercubes & High Performance Architectures	4 Days	$1195	$1295
290:	Effective Skills for Technical Managers	4 Days	$1195	$1295
296:	Project Planning & Control HANDS-ON	4 Days	$1195	$1295
340:	Software Project Management Tools & Techniques DEMO LABS	4 Days	$1295	$1395
348:	System Engineering for Integrated HW/SW Applications	4 Days	$1195	$1295
380:	Security for Computer and Communication Systems	4 Days	$1195	$1295
411:	Image Processing & Machine Vision DEMO LABS	4 Days	$1195	$1295
412:	Digital Signal Processing: Techniques & Applications DEMO LABS	4 Days	$1295	$1395
415:	Digital Signal Processing: Real-Time Hardware Design	4 Days	$1295	$1395
418:	Digital Control Systems: Design Techniques & Tools	4 Days	$1195	$1295

Course Hours & Completion Certificate:
Orientation (first day): 8:30am–9:00am
Class Hours: 9:00am–4:30pm (last day 9:00am–3:30pm)
Informal Discussion with Instructor: 4:30pm–5:30pm

Each attendee receives a diploma awarding a Continuing Education Unit (CEU) for each 10 hours of class participation.

BRING THESE COURSES TO YOUR FACILITY



▲ ***WHEN PRESENTED AT YOUR OWN FACILITY, THESE COURSES*** *provide even greater cost savings. Tuition is reduced, and travel expenses are eliminated, making it cost-effective to train entire teams.*

You'll Benefit From . . .
Proven, successful courses with expert instructors and comprehensive materials. For Hands-On and Demonstration Courses, we bring all of the in-class computers and electronic equipment to your facility.

A Major Advantage
Courses at your facility can be customized. Specific topics can be emphasized according to the needs of your organization, and timing can be adjusted to fit your schedule. By training an entire team, you'll accelerate the beneficial impact of new technology within your organization.

Experience Produces Results
Over the last 12 years, we have presented thousands of On-Site Courses for hundreds of organizations including: IBM, AT&T, GE, the US Navy, Tektronix, Hughes, HP, and TI. These and other references are available.

Take the First Step Today!
Courses at your facility are surprisingly easy and quick to organize. The first step can be taken by the technical staff, training department, or management.

For Courses at Your Facility Phone:
Anne-Lyse Heron or Reta Singer
On-Site Coordinators
(800) 421-8166 or
(213) 417-8888

We'll provide you with all the assistance and information you need to bring these high-impact courses to your organization.

Call today for a complete set of three brochures with details on 39 Technical Short Courses: (800) 421-8166.

Priority Code:
To facilitate enrollment, please give our registrar the Priority Code:

PCODE 5735



To Order Course Materials Separately:
Each participant receives a comprehensive set of course materials included in the course fee. If you are unable to attend, you may purchase the course materials separately.

Price: $275 per course (includes regular UPS shipping within USA)

Sales Tax: California residents please add 6% (6½% in Los Angeles County)

© Copyright Integrated Computer Systems 1988.
8805UB